

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

**КАФЕДРА СИСТЕМНОГО ПРОГРАМУВАННЯ І
СПЕЦІАЛІЗОВАНИХ КОМП'ЮТЕРНИХ СИСТЕМ**

«На правах рукопису»
УДК _____

«До захисту допущено»

Завідувач кафедри СПСКС

_____ В.П.Тарасенко
(підпис) (ініціали, прізвище)

“ ” _____ 2018р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 123 Комп'ютерна інженерія
(Спеціалізовані комп'ютерні системи)

на тему: Пошук попарних реберних циклів при модифікації GL-моделі

Виконав (-ла): студент (-ка) II курсу, групи КВ-63м
(шифр групи)

Ніколін Андрій Дмитрович
(прізвище, ім'я, по батькові) (підпис)

Науковий керівник Доцент кафедри СПіСКС, к.т.н., доцент Потапова К.Р.
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент ст. викладач каф. ПМА Мальчиков Володимир Вікторович
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студент _____
(підпис)

Київ – 2018 року

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет прикладної математики

Кафедра системного програмування і спеціалізованих комп'ютерних
систем

Рівень вищої освіти – другий (магістерський)
Спеціальність 123 Комп'ютерна інженерія
(Спеціалізовані комп'ютерні системи)

ЗАТВЕРДЖУЮ
Завідувач кафедри СПІСКС

В.П.Тарасенко
(підпис) (ініціали, прізвище)

«__» _____ 2018р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Ніколін Андрій Дмитрович
(прізвище, ім'я, по батькові)

1. Тема дисертації Пошук попарних реберних циклів при модифікації GL-моделі, науковий керівник дисертації

Доцент кафедри СПІСКС, к.т.н., доцент Потапова К.Р,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «22» березня 2018 р. №986-с

2. Термін подання студентом дисертації 11 травня 2018 р.

3. Об'єкт дослідження: моделі поведінки відмовостійких багатопроцесорних систем (ВБС), що дозволяють адекватно відобразити реакцію систем на появу відмов різних кратностей

4. Предмет дослідження: алгоритми формування, а також трансформації GL-моделей

5. Перелік завдань, які потрібно розробити

- Виконати огляд існуючих принципів побудови граф-логічних моделей
- Створити алгоритм формування реберних булевих функцій нової GL-моделі циклічного типу

- Дослідити попарні реберні цикли, що виникають при перетворенні базової GL-моделі в не базову
- Запропонувати метод зменшення кількості переборів при пошуку попарних реберних циклів

6. Перелік ілюстративного матеріалу: 10 рисунків, 8 таблиць.

7. Перелік публікацій:

- Потапова К.Р., Ніколін А.Д., Ахмедова Д.Н., Алгоритм пошуку попарних реберних циклів при перетворенні GL-моделей, Прикладна математика та комп'ютинг 2018, с.123-127
- Романкевич В.О., Ніколін А.Д., Корнейчик Б.А., Алгоритм перетворення базових GL-моделей, Прикладна математика та комп'ютинг 2018, с.141-144

8. Дата видачі завдання 5 вересня 2016 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
	Попередній розгляд магістерської дисертації на кафедрі	26.04.2018	

Студент

(підпис)

(ініціали, прізвище)

Науковий керівник дисертації

(підпис)

(ініціали, прізвище)

Реферат

Актуальність теми. Складні багатопроцесорні системи все більш широко використовуються у сучасному світі передовими країнами, що використовують сучасні технології проектування. В першу чергу це справедливо при побудові систем управління складними об'єктами у багатьох галузях промисловості. Втрата роботоздатності таких систем часом може привести до катастрофічних наслідків, як, наприклад, відмова системи управління атомною електростанцією, ракетами і літаками. Тому архітектура подібних багатопроцесорних систем спочатку повинна забезпечувати певний рівень надійності та відмовостійкості. Висока вартість реалізації великих систем управління привела до того, що велике значення на етапі проектування набуло моделювання роботи відмовостійких багатопроцесорних систем (ВБС), метою якого є оцінка надійності системи. Однак нерідко рішення цього завдання вимагає колосальних обчислювальних витрат і іноді не може бути виконано з необхідною точністю за заданий проміжок часу.

Сучасні методи аналізу і синтезу складних обчислювальних систем дають задовільні результати при визначенні продуктивності, функціональних можливостей та ін. Однак проаналізувати відмовостійкість системи надзвичайно складно через великий обсяг перебору її станів, що виникають при відмовах деяких елементів. Ефективні моделі для систем з відносно простою структурою, наприклад моделі на основі двополюсних графів і моделі, використовуючі "Марковські" ланцюги, не конструктивні для аналізу станів систем великої розмірності. Методи статистичного моделювання, що дозволяють зменшити цей перебір за рахунок отримання статистичних оцінок з обмеженою достовірністю, також

недостатньо ефективні через те, що описати критерії відмови надзвичайно складно.

Тому особливе місце серед безлічі моделей з практичних позицій займають графо-логічні моделі (GL-моделі), що відображають поведінку ВБС в потоці відмов, на основі статистичних експериментів з якими створено універсальний метод розрахунку надійності ВБС.

Об'єктом дослідження є моделі поведінки відмовостійких багатопроцесорних систем (ВБС), що дозволяють адекватно відобразити реакцію систем на появу відмов різних кратностей.

Предметом дослідження є алгоритми формування, а також трансформації GL-моделей.

Мета роботи. На основі відомих принципів побудови GL-моделей, створення алгоритму формування реберних булевих функцій нової GL-моделі циклічного типу. Дослідження попарних реберних циклів, що виникають при перетворенні базової GL-моделі в не базову.

Методи дослідження. В якості методів дослідження застосовувалися методи теорії булевих функцій, теорії графів, теорії алгоритмів.

Наукова новизна отриманих результатів визначається наступними положеннями:

1. Розроблено нову модель, відмінною рисою якої є простота формування реберних булевих функцій, а також рекурсивний алгоритм мінімізації моделі і показано, що для неї можна застосувати розроблений раніше метод перетворення шляхом проведення внутрішніх ребер зі своїми функціями.
2. Досліджено попарні реберні цикли і запропоновано алгоритм їх пошуку при перетворенні.

Практичну цінність представляє можливість використання результатів роботи при виконанні розрахунку характеристик надійності ВБС.

Апробація роботи. Основні положення і результати роботи були представлені і обговорювалися на десятій науковій конференції

магістрантів та аспірантів «Прикладна математика та комп'ютинг 2018».

Публікації. Результати наукових досліджень по темі магістерської дисертації відображені в 2 опублікованих роботах:

-;
-

Структура та об'єм роботи.

У **вступі** обґрунтовується актуальність теми роботи, вказується мета роботи і сфера застосування результатів дослідження.

У **розділі 1** наведені основні визначення і поняття, а також загальні відомості про проблему розрахунку надійності багатопроцесорних систем.

У **розділі 2** описані відмовостійкі багатопроцесорні системи, проблеми їх проектування; описуються різні моделі, що відображають реакцію системи на появу тих чи інших відмов.

У **розділі 3** описується відомий метод побудови канонічної GL-моделі поведінки ВБС, стійкої до появи відмов заданої кратності. Представлений метод побудови нової GL-моделі поведінки ВБС і алгоритм її мінімізації.

У **розділі 4** аналізується можливість використання двох шляхів трансформації GL-моделей, зокрема, заміни деяких ребрових функцій і додавання внутрішніх ребер з метою збереження адекватності моделі при трансформації системи в небазових. Аналізуються так звані попарні реберні цикли, що виникають в моделі при її перетворенні шляхом введення внутрішніх ребер.

У **висновку** зроблені загальні заключення, виконано аналіз отриманих результатів.

У **додатку** – копії графічного матеріалу і довідка про впровадження.

Робота містить ## сторінок друкованого тексту, малюнки, таблиці та список використаної літератури.

Ключові слова: ВІДМОВОСТІЙКІ БАГАТОПРОЦЕСОРНІ СИСТЕМИ, БУЛЕВІ ФУНКЦІЇ, ГРАФО-ЛОГІЧНА МОДЕЛЬ.

Реферат

Актуальность темы. Сложные многопроцессорные системы все более широко используются в современном мире передовыми странами, которые используют современные технологии проектирования. В первую очередь это справедливо при построении систем управления сложными объектами во многих отраслях промышленности. Потеря работоспособности таких систем подчас может привести к катастрофическим последствиям, как, например, отказ системы управления атомной электростанцией, ракетами и самолетами. Поэтому архитектура подобных многопроцессорных систем изначально должна обеспечивать определенный уровень надежности и отказоустойчивости. Высокая стоимость реализации крупных систем управления привела к тому, что большое значение на этапе проектирования приобрело моделирование работы отказоустойчивых многопроцессорных систем (ОМС), целью которого является оценка надежности системы. Однако нередко решение этой задачи требует колоссальных вычислительных затрат и иногда не может быть выполнено с необходимой точностью за заданный промежуток времени.

Современные методы анализа и синтеза сложных вычислительных систем дают удовлетворительные результаты при определении производительности, функциональных возможностей и др. Однако проанализировать отказоустойчивость системы чрезвычайно сложно из-за большого объема перебора ее состояний, возникающих при отказах некоторых элементов. Эффективные модели для систем с относительно простой структурой, например модели на основе двухполюсных графов и модели, использующие "Марковские" цепи, не конструктивны для анализа состояний систем большой размерности. Методы

статистического моделирования, позволяющие уменьшить этот перебор за счет получения статистических оценок с ограниченной достоверностью, также недостаточно эффективны из-за того, что описать критерии отказа чрезвычайно сложно.

Поэтому особое место среди множества моделей с практических позиций занимают графо-логические модели (GL-модели), отражающие поведение ОМС в потоке отказов, на основе статистических экспериментов с которыми создан универсальный метод расчета надежности ОМС.

Объектом исследования являются модели поведения отказоустойчивых многопроцессорных систем (ОМС), которые позволяют адекватно отражать реакцию систем на появление отказов различной кратности.

Предметом исследования является алгоритмы формирования, а так же трансформации GL-моделей.

Цель работы. На основе известных принципов построения GL-моделей, создание алгоритма формирования рёберных булевых функций новой GL-модели циклического типа. Исследование попарных рёберных циклов, возникающих при преобразовании базовой GL-модели в небазовую.

Методы исследования. В качестве методов исследования применялись методы теории булевых функций, теории графов, теории алгоритмов.

Научная новизна полученных результатов определяется следующими положениями:

3. Разработана новая модель, отличительной особенностью которой простота формирования рёберных булевых функций, а также рекурсивный алгоритм минимизации модели и показано, что для нее применимы разработанный ранее метод преобразования путем проведения внутренних ребер со своими функциями.
4. Исследованы попарные реберные циклы и предложен алгоритм их отыскания при преобразовании.

Практическую ценность представляет возможность использования результатов работы при выполнении расчета надежностных характеристик ОМС.

Апробация работы. Основные положения и результаты работы были представлены и обсуждались на

Публикации. Результаты научных исследований по теме магистерской диссертации отражены в 2 опубликованных работах:

-
-

Структура и объем работы.

Во **введении** обосновывается актуальность темы работы, указывается цель работы и сфера применения результатов исследования.

В **разделе 1** приведены основные определения и понятия, а также общие сведения о проблеме расчета надёжности многопроцессорных систем.

В **разделе 2** описаны отказоустойчивые многопроцессорные системы, проблемы их проектирования; описываются разные модели, отражающие реакцию системы на появление тех или иных отказов.

В **разделе 3** описывается известный метод построения канонической GL-модели поведения ОМС, устойчивой к появлению отказов заданной кратности. Представлен метод построения новой GL-модели поведения ОМС и алгоритм ее минимизации.

В **разделе 4** анализируются возможности использования двух путей трансформации GL-моделей, в частности, замены некоторых реберных функций и добавление внутренних ребер с целью сохранения адекватности модели при трансформации системы в небазовую. Анализируются так называемые попарные реберные циклы, возникающие в модели при ее преобразовании путем введения внутренних ребер.

В **заключении** сделаны общие выводы, выполнен анализ полученных результатов.

В **приложении** – копии графического материала и справка о внедрении.

Работа содержит 89 страниц печатного текста, рисунки, таблицы и список использованной литературы.

Ключевые слова: ОТКАЗОУСТОЙЧИВЫЕ
МНОГОПРОЦЕССОРНЫЕ СИСТЕМЫ, БУЛЕВЫ ФУНКЦИИ,
ГРАФО-ЛОГИЧЕСКАЯ МОДЕЛЬ.

Abstract

Actuality of theme. Complex multiprocessor systems are widely used in the modern world by countries using modern design technologies. In the first place, they are used during the design process of complex object management systems in many industries. Loss of the working proficiency of such systems may lead to catastrophic consequences, such as the failure of the nuclear power plant control system, missiles and airplanes. Therefore, the architecture of such multiprocessor systems should initially provide a certain level of reliability and fault tolerance. The high cost of implementation of large control systems has led to the importance of designing the fault-tolerant multi-processor systems (FTMS), the purpose of which is to assess the reliability of the system. However, quite often, solving this problem requires enormous computing costs and sometimes can not be performed with the necessary accuracy for a given time interval.

Modern methods of analysis and synthesis of complex computing systems give satisfactory results in determining the productivity, functionality, etc. However, to analysis of the fault-tolerance of the system is extremely difficult due to the large amount of overview of its states that arise in case of failures of some elements. Effective models for systems with a relatively simple structure, such as models based on bipolar graphs and models using Markov chains, are not constructive for the analysis of states of systems of large dimension. The methods of statistical simulation, which allow to reduce this survey by obtaining statistical estimates with limited reliability, are also not sufficiently effective due to the fact that describing the criteria for failure is extremely difficult.

Therefore, a special place among the many models of practical positions occupy graph-logic models (GL-models) that reflect the behavior of the

FTMS in the flow of failures, based on statistical experiments, which created a universal method for calculating the reliability of the FTMS.

The object of the study is the behavior patterns of fault-tolerant multiprocessor systems (FTMS) that allow to adequately reflect the reaction of systems to the occurrence of failures of various multiples.

The subject of the study is algorithms for the formation, as well as the transformation of the GL model.

The purpose of the work. On the basis of known principles of construction of GL-models, the creation of algorithm for the formation of ribs Boolean functions of a new GL-model of cyclic type. Investigation of pairwise finer cycles that arise when transforming a basic GL model into a non-basic.

Research methods. As methods of research were used methods of the theory of boolean functions, graph theory, theory of algorithms were used.

The scientific novelty of the results obtained is determined by the following provisions:

5. A new model is developed, the distinctive feature of which is the simplicity of the formation of ribs of boolean functions, as well as the recursive algorithm for minimizing the model, and it is shown that for it it is possible to apply the previously developed method of transformation by holding internal ribs with their functions.
6. Two-dimensional finite cycles have been investigated and the algorithm of their search for transformation has been proposed.

Practical value is represented by the possibility of using the results of work when performing the calculation of reliability characteristics of the FTMS.

Work approbation. The main provisions and results of work were presented and discussed at the tenth scientific conference of masters and postgraduates "Applied Mathematics and Computing 2018".

Publications. The results of scientific research on the theme of the master's dissertation are reflected in 2 published works:

-;
-

Structure and volume of work.

The **introduction** substantiates the relevance of the topic of work, indicates the purpose of the work and the scope of the results of the study.

Chapter 1 provides basic definitions and concepts, as well as general information about the problem of calculating the reliability of multiprocessor systems.

Chapter 2 describes fault-tolerant multiprocessor systems, problems of their design; describes different models that reflect the reaction of the system to the occurrence of certain failures.

Chapter 3 describes a well-known method for constructing a canonical GL-behavior model of VBS that is stable until failure of a given multiplicity. The method of constructing a new GL-model of the behavior of the VBS and its minimization algorithm is presented.

Chapter 4 analyzes the possibility of using two ways of transforming GL models, in particular, replacing some rib functions and adding internal edges in order to maintain the adequacy of the model when transforming the system into non-base ones. The so-called pairwise finer cycles, which arise in the model during its transformation by the introduction of internal ribs, are analyzed.

In **the conclusion** an analysis of the results is carried out.

In **the appendix** – copies of the graphic material and a certificate of introduction.

The work contains ## pages of printed text, drawings, tables and list of used literature.

Keywords: FAULT-TOLERANT MULTI-PROCESSOR SYSTEMS, BOOLEAN FUNCTIONS, GRAPH-LOGIC MODEL.

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ І ПОЗНАЧЕНЬ	16
ВСТУП	18
1. БАГАТОПРОЦЕСОРНІ СИСТЕМИ ТА ЇХ НАДІЙНІСТЬ	21
2. МОДЕЛІ ПОВЕДІНКИ ВБС В ПОТОЦІ ВІДМОВ	28
2.1 Використання графів для розрахунку надійності ВБС	28
2.2 Реберні функції GL-моделей	31
2.3 Висновки.....	38
3. ПОБУДОВА БАЗОВИХ GL-МОДЕЛЕЙ.....	40
3.1 Реберні функції базової GL-моделі і їх формування.....	40
3.1.1 2-відмовостійкі ВБС	40
3.1.2 GL-моделі загального виду.....	46
3.1.3 Приклад	53
3.2 Реберні функції на основі КНФ.....	62
3.2.1 Алгоритм	66
3.2.2 Приклад	68
3.2.3 Мінімізація GL-моделі системи, стійкої до відмов довільної кратності.....	70
3.2.4 Приклад мінімізації.....	77
3.2.5 Висновки.....	82
4. ШЛЯХИ ПЕРЕТВОРЕННЯ ЦИКЛІЧНИХ GL-МОДЕЛЕЙ	84
4.1 Внутрішні ребра в графі GL-моделі	84
4.1.1 Реберні функції	85
4.1.2 Додаткові ребра	86
4.2 Заміна реберних функцій GL-моделі.....	89
4.2.1 Зміна реберних функцій.....	90
4.2.2 Додаткові ребра	91
4.3 Перетворення МВР-моделей	94
4.4 Попарні реберні цикли при перетворенні GL-моделей	103
4.1 Висновки.....	112
ВИСНОВКИ	114
ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	116

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ І ПОЗНАЧЕНЬ

GL-модель – графо-логічна модель поведінки ВМС в потоці відмов

ВБС – відмовостійка багатомодульна (багатопроесорна) система

ВОС – відмовостійка обчислювальна система

k-ВБС – багатомодульна (багатопроесорна) система, стійка до k відмов

k-ВБС – обчислювальна система, стійка до k відмов

Базова GL-модель - модель ВБС, стійка до відмов, що не перевищує заданої кратності

Д.н.ф. – диз'юнктивна нормальна форма

К.н.ф. – кон'юнктивна нормальна форма

ВСТУП

Для управління складними об'єктами в багатьох сучасних галузях промисловості і науки широко застосовуються складні багатопроцесорні системи. Втрата працездатності таких систем часом може призвести до катастрофічних наслідків, як, наприклад, відмова системи управління атомною електростанцією, ракетами і літаками. Тому архітектура подібних багатопроцесорних систем з самого початку повинна забезпечувати певний рівень відмовостійкості. Висока вартість реалізації великих систем управління привела до того, що велике значення набуло моделювання роботи відмовостійких багатопроцесорних систем (ВБС), що дозволяє на етапі проектування дати оцінку надійності системи. Однак нерідко рішення цього завдання вимагає колосальних обчислювальних витрат і іноді не може бути виконано з необхідною точністю за заданий проміжок часу.

Сучасні методи аналізу і синтезу складних обчислювальних систем дають задовільні результати при визначенні продуктивності, функціональних можливостей та ін. Однак проаналізувати відмовостійкість системи надзвичайно складно через великий обсяг перебору її станів, що виникають при відмовах деяких елементів. Ефективні моделі для систем з відносно простою структурою, наприклад моделі на основі двополюсних графів і моделі, що використовують "Марковські" ланцюги, неконструктивні для аналізу станів систем великої розмірності. Методи статистичного моделювання, що дозволяють зменшити цей перебір за рахунок отримання статистичних оцінок з обмеженою достовірністю, також недостатньо ефективні через те, що описати критерії відмови надзвичайно складно.

Серед безлічі моделей ВБС особливе місце займають графологічні моделі (GL-моделі), що відображають поведінку ВБС в потоці відмов. Що дозволяють спростити процедуру визначення працездатності ВБС.

Метою роботи: на основі відомих принципів побудови GL-моделей, створення алгоритму формування реберних булевих функцій нової GL-моделі циклічного типу та дослідження попарних реберних циклів, що виникають при перетворенні базової GL-моделі в небазову.

1. БАГАТОПРОЦЕСОРНІ СИСТЕМИ ТА ЇХ НАДІЙНІСТЬ

У багатьох прикладних задачах виникає необхідність визначення різних показників надійності систем зі складною структурою. Для даної задачі використовуються графо-логічні моделі (GL - моделі).

Моделі поведінки відмовостійких багатомодульних (багатопроцесорних) систем (ВБС) в потоці відмов дозволяють ще на етапах проектування оцінити такі кількісні характеристики, як надійність та ефективність цих систем. Без належного дослідження, без проведення кількісного аналізу неможлива розробка конкретних та ефективних заходів щодо забезпечення високої надійності технічних систем.

Надійність є однією з основних інженерних проблем. Проблемою надійності займалися завжди з тих пір, як з'явилася техніка. Ненадійні вироби ніколи нікому не були потрібні. Давно вже було зрозуміло, що надійність пов'язана з надмірністю. У зв'язку з цим в інженерних розрахунках в різних областях техніки широко використовуються необхідні коефіцієнти запасу.

Однак за останні 25-30 років проблема надійності технічних систем і елементів, що входять до неї, сильно загострилася. Це обумовлено головним чином наступними причинами:

1. Зростанням складності сучасних технічних систем, що включають $10^4 \div 10^6$ окремих елементів;
2. Інтенсивністю режимів роботи системи або окремих її частин: при високих температурах, високому тиску, високих швидкостях;
3. Складністю умов, в яких експлуатується технічна система, наприклад: низькі або високі температури, високі вологість, вібрації, прискорення і радіація і т. п.;

4. Вимогами до якості роботи системи: високі точність, ефективність і т. п.;

5. Підвищенням відповідальності функцій, виконуваних системою; високою технічною та економічною ціною відмови;

6. Повною або частковою автоматизацією і виключенням безпосередньої участі людини при виконанні технічною системою її функцій, виключенням безперервного спостереження і контролю з боку людини.

Однією з головних причин загострення уваги до проблеми надійності є зростання складності технічних систем.

В якості прикладів можна вказати:

літаки-бомбардувальники США мали в 1945 р. (типи Б-17 та Б-29) до 2000 елементів, в 1960 р. (тип Б-58) до 95 000, а в 1965 р. (тип В-70) до 150 000 елементів;

обчислювальний пристрій AN/FSQ-7, що входить в систему виявлення ПРО США, включає більш ніж 50 000 електронних ламп, 170 000 напівпровідникових діодів, 547 000 опорів, 189 000 конденсаторів та велику кількість інших елементів;

система керування американської міжконтинентальної балістичної ракети «Атлас» включає близько 300 000 елементів, а система управління ракети «Найк» — більш ніж $1,5 \cdot 10^6$ окремих елементів;

сучасні електронні цифрові обчислювальні машини мають близько 10 000-15 000 напівпровідникових тріодів і в 6-15 разів більшу кількість інших електричних компонентів (опорів, конденсаторів і т. д.);

система автоматичного управління сучасними металургійними або хімічними процесами включає до 200-300 окремих контрольних та регулюючих приладів, кожен з яких містить до 100-200 окремих компонентів.

Інтенсивність режимів роботи технічних систем характеризується застосуванням пари при високих температурах та тисках, застосуванням високих температур в камерах реактивних двигунів, застосуванням високих швидкостей в сучасних турбореактивних двигунах, гіроскопах та інших елементах і приладах.

Складність умов, в яких можуть експлуатуватися сучасні технічні системи, характеризується роботою в широких діапазонах температур від -70° до $+70^{\circ}$ С, наявністю вакууму, високою (98-100%) вологістю, вібраціями з великою амплітудою і широким спектром частот, наявністю лінійних прискорень до 10 300 (1000) і навіть 20 000 g, наявністю високої сонячної і космічної радіації.

Це призводить до того, що ймовірності виникнення відмов можуть зрости в 25-100 або навіть 500-1000 разів в порівнянні з ймовірністю відмов при роботі технічних систем в умовах лабораторій.

Складність апаратури і важкі експлуатаційні умови ускладнюють контроль за справністю апаратури, що входить в технічну систему, що не дає можливості своєчасно виявити процеси, що призводять до відмови, і попередити її появу.

Відповідальність функцій, які виконуються сучасними технічними системами, пов'язана з тим, що відмова її призводить до великих технічних та економічних втрат. У ряді випадків це може викликати катастрофічні наслідки, наприклад:

відмова елемента вартістю в 5 доларів викликала в США невдачу при запуску супутника вартістю $\sim 8 \cdot 10^6$ доларів;

відмова в роботі теплового екрана викликала аварію вартістю в $20 \cdot 10^6$ доларів;

збиток через відмову апаратури автоматичного управління виробничим процесом в хімічній промисловості в сотні разів перевищує

вартість самої апаратури управління та може призвести до загибелі устаткування чи людей;

відмова релейного захисту в енергосистемі північно-східної частини США викликала зупинку енергоживлення ряду штатів та призвела до збитків $\sim 500 \cdot 10^6$ долларів.

Проблема забезпечення надійності пов'язана з усіма етапами створення виробу і всім періодом його практичного використання.

Надійність виробу закладається в процесі його конструювання і розрахунку та забезпечується в процесі його виготовлення шляхом правильного вибору технології виробництва, контролю якості вихідних матеріалів, напівфабрикатів і готової продукції, контролю режимів і умов виготовлення.

Розрахунок надійності став обов'язковим інженерним розрахунком на всіх етапах розробки, створення та використання технічних засобів.

На етапі ескізного проектування розрахунок надійності проводиться з метою прогнозування очікуваних показників надійності. Цей розрахунок необхідний для того, щоб обґрунтувати ідею пропонованого варіанту системи та показати, що очікувані показники надійності будуть не гірші необхідних.

На етапі технічного проектування результати розрахунків надійності використовуються для обґрунтування вибору технічних засобів, що входять в систему, а також для вибору способів резервування, глибини та способів контролю і діагностики, обґрунтування структури системи, вимог до надійності комплектуючих елементів і програмного забезпечення, а також до інших факторів, що впливає на надійність.

Розрахунки надійності на цьому етапі проводяться, як правило, багаторазово, так як можливі багаторазові зміни архітектури системи в процесі проектування.

На етапі випробувань системи розрахунок надійності проводиться з метою визначення відповідності показників надійності випробуваної системи заданим вимогам. Важливість розрахунків надійності на даному етапі визначається тим, що з випробування надійності великих системи вона не може бути пред'явлена в великій кількості екземплярів. Часто вона пред'являється в скороченому варіанті і потрібно визначити надійність системи на підставі обмеженої інформації зі сфери випробувань. Тільки поєднання натурних випробувань системи з розрахунками, заснованими на використанні всієї наявної інформації про надійність системи та її складових частин, дозволяє вирішити задачу оцінки надійності великої системи на етапі її випробувань.

На етапі експлуатації системи розрахунки надійності використовуються для вибору та обґрунтування складу і об'єму запасних виробів, необхідних для заміни відмовивших, а також для обґрунтованого планування профілактичного обслуговування.

Таким чином, на всіх етапах створення та використання ВБС розрахунки надійності мають важливе значення. Перед початківцем в вивчення методів розрахунків надійності відкриваються два шляхи. Перший - менш трудомісткий і досить привабливий шлях - заучування готових прийомів та методів розрахунку і вироблення навичок швидкого відшукування за довідником найбільш придатної розрахункової формули. Але такий шлях таїть в собі небезпеку заучування готових формул без розуміння їхнього змісту і неминуче призведе до того, що фахівець з розрахунками рано чи пізно зустрінеться з ситуацією, що не передбачена довідником за розрахунками. Другий шлях вивчення методів розрахунку надійності полягає в знаходженні загальних закономірностей, принципів засад, які можуть бути покладені в основу розрахунків. Якщо ці закономірності визначені хоча б у першому наближенні, то при виникненні конкретної задачі розрахунку вибір розрахункової формули

стає осмисленим. При відсутності готової розрахункової формули ставиться завдання пошуку її на підставі загальних закономірностей. Ці два шляхи не виключають один одного. Навпаки, вони повинні доповнювати один одного. Не слід кожен раз робити спробу відшукувати «свою» розрахункову формулу, якщо вона існує як довідкова. Однак не можна орієнтуватися тільки на довідникові формули, нехтуючи первоосновами розрахунків.

2. МОДЕЛІ ПОВЕДІНКИ ВБС В ПОТОЦІ ВІДМОВ

2.1 Використання графів для розрахунку надійності ВБС

Більшість територіально розподілених об'єктів (систем) мають мережеву структуру. До них відносяться й такі стародавні комунікаційні системи, як мережі доріг, й такі сучасні системи інформатики, як обчислювальні мережі. До цього ж класу систем відносяться залізничні мережі, мережі авіаційних ліній, системи зв'язку і системи передачі даних, електроенергетичні мережі - від магістральних до локальних розподільних, мережі трубопроводів (для транспортування нафти, газу, води).

Для них характерним є те, що всі включені в їх склад об'єкти можуть бути поділені з досить загальних позицій на два класи: термінальні та транзитні пункти (місця відправлення матеріальних, енергетичних або інформаційних потоків, а також місця їх переадресації та отримання) та комунікаційні канали - лінії зв'язку між різними термінальними й транзитними пунктами. Залежно від фізичної природи цих систем комунікаційні канали можуть мати або цілком певну матеріальну структуру, наприклад нафтопроводи, залізниці, системи електропередач і т.п., а можуть бути чисто умоглядними, наприклад траси авіаліній та морських шляхів, напрямки радіозв'язку тощо. Однак і в тому і в іншому випадку всі ці системи мають безліч схожих рис. Саме тому для математичного опису таких, настільки особистих по природі, призначенням і принципом функціонування систем виявляється вельми зручним використовувати математичні моделі, які спираються на теорію графів.

З використанням математичного апарату цієї теорії можна вирішувати завдання аналізу надійності.

По-перше, мова теорії графів виявляється вельми природним для опису структур багатьох технічних систем. В основному це стосується різних комунікаційних систем.

По-друге, уявлення структури системи у вигляді графа є першим етапом складання математичної моделі (наприклад, графи переходів служать безпосередньою основою для складання відповідних систем рівнянь).

По-третє, формулювання завдання в термінах графів дозволяє використовувати для її вирішення вже наявний математичний апарат теорії графів.

Довгий час теорія графів була тільки чисто «мовним» засобом. Однак останнім часом вона починає використовуватися в роботах по надійності все більш конструктивним чином.

Нагадаю, що графом називається сукупність вершин (вузлів) та ребер, пов'язаних між собою. Строго кажучи, графом називається абстрактна математична система, що складається з двох множин - вершин і ребер і відображення безлічі вершин в безліч ребер (відображення інцидентності). Таким чином, різні вузли графа пов'язані між собою (не обов'язково всі) ребрами. У кожен вузол може входити або з вузла може виходити кілька ребер. У цьому випадку говорять про безліч ребер, інцидентних даному вузлу графа. Кожному ребру інцидентні два вузла, розташовані на його кінцях. Самі ребра можуть бути простими і кратними, спрямованими і ненаправленими. Кратні ребра - це просто кілька (як правило, ідентичних) ребер, інцидентних одній і тій же парі вершин. В іншому випадку ребра називаються простими. Сказане стосується в першу чергу спрямованих ребер,

які також називаються іноді дугами. Якщо по ненаправленому ребру допускається ідентичний зв'язок від однієї вершини до іншої, то для спрямованого ребра (дуги) один з вузлів є джерелом, а інший - стоком і передача потоку в зворотному напрямку по дузі неможлива. Якщо в кінцевому графі, тобто в графі, що включає кінцеве число вершин, з будь-якої вершини через кінцеве число ребер (або дуг) можна потрапити в будь-яку іншу вершину, то говорять про зв'язність графа. Дуже часто в прикладних задачах виникає поняття зв'язності між собою двох обернених вершин графа (одна з яких називається зазвичай початковою, а інша - кінцевою вершиною або вхідним та вихідним полюсами відповідно).

Подивимося, як ці абстрактні поняття теорії графів знаходять своє переломлення в різних прикладних задачах. Згадаю лише ті поняття, які можуть здатися неочевидними з точки зору фізичного тлумачення. Ненаправлене ребро - це, наприклад, траса авіалінії, так як в принципі літаки можуть літати в обидві сторони. Однак в ряді досліджень можна говорити про наявність двох спрямованих у різні сторони дуг. Зазвичай трубопровідні системи працюють на передачу потоку в цілком певному напрямку, тому при певних дослідженнях їх зручно розглядати як спрямовані ребра. Але в той же час можливі реверсивні режими (наприклад, перемички в нафто- та газопроводах), і тоді може виявитися доцільним деякі з ребер вважати ненаправленими. Прикладом пов'язаного графа є граф, що зв'язує світову систему авіаліній, що сполучає всі країни світу: пасажир в принципі (і причому за порівняно невелику кількість пересадок, тобто використовуючи відносно невелике

число ребер графа) може з будь-якого аеропорту однієї країни потрапити в будь-який аеропорт іншої країни.

По теорії графів є велика кількість літератури. Говорячи про теорії графів, мається на увазі класичні теоретико-множинні моделі, залишаючи поки поза увагою так звані, лінійні графи та різні спеціальні методи, пов'язані з ними.

2.2 Реберні функції GL-моделей

Відмовостійкі багатопроцесорні обчислювальні і керуючі системи з кожним роком набувають все більшого поширення. Це пов'язано з розширюючоюся областю їх застосування, в першу чергу в системах управління складними та відповідальними об'єктами (літаки, ракети, космічні об'єкти, складні технологічні процеси, фінансові розрахунки та ін.). При цьому для обчислювальних систем характерні велика кількість процесорів з одного боку та регулярність структури їх з'єднань - з іншого. Це полегшує певною мірою їх проектування й розрахунок необхідних параметрів.

Для систем управління навпаки - порівняно менша кількість процесорів (десятки, сотні, іноді тисячі) і відносна нерегулярність структури. Останнє пов'язано, як правило, з великою кількістю джерел інформації різного типу (датчиків) і різнотипністю процесорів. Для нас інтерес представляє так само той факт, що різні частини систем, що управляють різними ділянками об'єкта (простіше кажучи, підсистеми), повинні бути захищені від відмов по-різному.

Всі ці системи, як правило, самі себе тестують та реконфігурують, і незважаючи на їх відмінність, підходи до вирішення завдань забезпечення вимог відмовостійкості багато в чому подібні. Тому ми збережемо в подальшому поширену абревіатуру ВБС (відмовостійкі

багатопроцесорні системи), хоча в першу чергу будемо мати на увазі системи управління об'єктами.

ВБС, стійку до відмов певної кратності, будемо називати базовою. Сказане означає, що число відмов елементів такої системи не повинна перевищувати певної фіксованої величини, інакше система буде не в змозі продовжувати функціонування з тими ж параметрами.

Реальні ВБС в цілому не підкоряються цьому правилу: ВБС може вийти з ладу при виникненні k певних відмов i , в той же час, бути стійкою до s іншим відмов при $s > k$. Цей факт змушує нас підкреслити два моменти:

Наведене визначення найчастіше відноситься до деякої частини системи, тобто ми будемо говорити про базові підсистеми ВБС;

При досить великій кількості модулів ВБС стає очевидною необхідність моделювання реакції ВБС на ту чи іншу безліч відмов її компонент.

У другому випадку ми будемо говорити про моделювання поведінки ВБС в потоці відмов. Це доводиться робити при розрахунках параметрів ВБС, в першу чергу при розрахунках надійності. Відразу ж відзначимо особливість відмовостійких багатопроцесорних систем управління: дуже часто її підсистеми не відповідають вимогам базових. Іншими словами, дослідження ВБС не можна обмежувати лише базовими системами.

Тому побудову моделей поведінки ВБС в потоці відмов й відповідний аналіз самих підсистем та систем будемо проводити в два етапи: спочатку для базових та, в подальшому, для більш складних систем й моделей.

При проектуванні відмовостійких багатопроцесорних систем необхідно вирішувати цілий ряд задач, в тому числі передбачати надлишкові програмно-апаратні засоби для тестування та виявлення

відмовивших модулів, а потім виконувати процедури реконфігурації системи, що полягають в інформаційній ізоляції відмовивших модулів, збереженні даних, використанні резерву (як структурного, так і тимчасового) і відновлення обчислювального процесу.

Очевидно, що для оптимального вибору апаратних і алгоритмічних ресурсів, що забезпечують реконфігурацію системи, необхідна досить точна оцінка параметрів надійності ВБС на ранніх етапах її проектування. У той же час відомі методи розрахунку параметрів надійності в разі ВБС дають неточну оцінку. Спроби наблизити розрахункові оцінки до реальних стикаються з непереборними обчислювальними труднощами, що пов'язані з величезним перебором можливих станів системи.

Сказане вище стосується, наприклад, розрахунків надійності ВБС, що засновані на використанні моделі двополюсного графа. Критерієм працездатності системи в такій моделі є існування шляху між двома виділеними полюсами графа. Аналіз проводиться за умови випадкового існування ребер графа. При великій складності аналізованої системи завдання як побудови самого графа, так і перебору всіх можливих шляхів в двополюсному графі переходить в клас неконструктивних завдань.

У деяких випадках (наприклад, для мережевих структур) в якості моделі для розрахунку надійності може бути обраний багатополісний граф, і критерієм надійності є зв'язність графа (система вважається працездатною тільки тоді, коли кожен вузол графа має зв'язок з будь-яким іншим вузлом). Завдання полягає у визначенні ймовірності події, що полягає в тому, що, по хоча б, одне дерево ненаправленого графа працездатне. При цьому комбінаторна задача перебору всіх дерев для дуже складних мереж призводить, як і в випадку з двополюсними графами, до значних обчислювальних труднощів.

Методи статистичного моделювання, що дозволяють обійти проблему перебору дуже великого числа станів досліджуваної системи, за рахунок отримання статистичних оцінок з заданою вірогідністю, в значній мірі втрачають свою перевагу через, як правило, надзвичайно складний опис використовуваних критеріїв, в якості яких застосовуються структурні функції, що характеризують працездатність системи.

Наведені приклади підтверджують актуальність завдання побудови нових моделей поведінки ВБС, які дозволили б зробити аналіз станів системи більш доступним і наглядним, і в результаті спростили б розрахунки та отримали досить точну оцінку параметрів надійності.

Відзначимо, що для ефективного аналізу поведінки систем, що мають велике число станів, вельми важливим є компактне подання критерію, відповідно до якого здійснюється оцінка поведінки системи в певних умовах.

Основна ідея нового підходу полягає в побудові моделі, що поєднує використання властивостей графів і булевих функцій. Таку модель в подальшому будемо називати графо-логічною. Нова модель передбачає використання для відображення поведінки системи відповідності між зв'язністю деякого графа і обчислюваними для кожного стану системи значеннями певним чином обраних булевих функцій. Однак на відміну від відомих моделей, що використовують графи, в даному випадку граф, що використовується може мати набагато більш просту структуру, ніж структура моделюємої системи. Цього вдається досягти за рахунок «перенесення» складності на булеві функції, якими позначаються ребра графа. При цьому критерій, який використовується для визначення стану системи, може бути зроблений

досить простим, якщо вдається обрати регулярну форму подання булевих функцій в графо-логічній моделі.

Опишемо коротко основні принципи побудови графо - логічної моделі. Відзначимо, що головна ідея полягає в створенні можливості змінювати структуру графа відповідно до вектору стану елементів системи, що досягається шляхом приписування ребрам графа певних булевих функцій.

При аналізі надійності різних технічних систем, перш за все, вводиться критерій відмови. На практиці критерієм відмови технічної системи є порушення здатності цієї системи виконувати своє призначення або невідповідність значень вихідних параметрів деяким вимогам. При створенні математичної моделі структури технічної системи критерій відмови зазвичай визначається через стан елементів, що складають дану систему. У цьому випадку кожен з елементів системи передбачається найпростішим, тобто вважається, що він може перебувати лише в двох станах: працездатності та відмови. Стан системи визначається сукупністю станів її елементів: інакше кажучи, критерій відмови дозволяє усю множину станів елементів системи розділити на дві підмножини - одна з них характеризується станом працездатності системи, а інша - стан відмови системи. У цьому випадку аналіз надійності системи зі складною структурою зводиться до подання системи у вигляді деякого еквівалентного елемента.

За допомогою підходу, заснованого на використанні графо-логічної моделі (GL - моделі) поведінки ВБС, що дозволяє зробити аналіз стану системи більш наочним і доступним. Для відображення поведінки системи за допомогою GL-моделі проводиться аналіз зв'язності деякого графа, існування ребер якого залежить від значень

певним чином обраних булевих функцій. Побудова GL-моделі розглянуто для випадку, коли аналізована система має n модулів і стає непрацездатною за умови відмови не менше, ніж $k + 1$ модулів ($k < n$). Таку ВБС назовемо k -ВБС. ВБС, стійку до відмов певної кратності, будемо називати базовою. Зрозуміло, що безліч можливих станів системи складається з 2^n різних станів. Залежно від конкретної структури системи все ця множина станів може бути розбита на дві підмножини, що відповідають станам працездатності та відмови системи в цілому.

Реальні ВБС в цілому не підкоряються цьому правилу: ВБС може вийти з ладу при виникненні k певних відмов і, в той же час, бути стійкою до s інших відмов при $s > k$. Цей факт змушує нас підкреслити два моменти:

1. Наведене визначення найчастіше відноситься до деякої частини системи, тобто ми будемо говорити про базові підсистеми ВБС;
2. При досить великій кількості модулів ВБС стає очевидною необхідність моделювання реакції ВБС на ту чи іншу безліч відмов її компонент.

В другому випадку ми будемо говорити про моделювання поведінки ВБС в потоці відмов. Це доводиться робити при розрахунках параметрів ВБС, в першу чергу при розрахунках надійності. Відразу ж відзначимо особливість відмовостійких багатопроцесорних систем управління: дуже часто її підсистеми не відповідають вимогам базових. Іншими словами, дослідження ВБС не можна обмежувати лише базовими системами.

Довільна ВБС є композицією різних k -ВБС. GL - модель для k -ВБС будується наступним чином. Обирають і ставлять у відповідність k -ВБС якийсь зв'язний неорієнтований граф G , ребра якого позначаються певним чином обраними булевими функціями $f_1(x_1, \dots, x_n), \dots, f_i(x_1, \dots, x_n)$. Тут кожна змінна x_i ($i=1..n$) є індикаторної функцією,

що позначає стан i -го елемента системи, тобто $x_i = 0$, якщо i -й елемент системи працездатний; $x_i = 1$ - в іншому випадку. Ребро, позначене функцією $f_i(x_1, \dots, x_n)$ залишається в графі G , якщо $f_i(x_1, \dots, x_n) = 0$ і видаляється з нього, якщо $f_i(x_1, \dots, x_n) = 1$.

Реберні функції представлені у вигляді своїх скорочених ДНФ і має вигляд: $f_i = C_{i1} \vee C_{i2} \vee \dots \vee C_{ip_i}$, де C_{ij} - j -та елементарна кон'юнкція в i -тій реберній функції.

Якщо вибрати функції $f_1(x_1, \dots, x_n), \dots, f_i(x_1, \dots, x_n)$ таким чином, щоб граф G втрачав зв'язність тільки при рівності нулю не менше $k + 1$ будь-яких з n змінних, то відмову n -модульної системи можна інтерпретувати, як втрату зв'язності графа G . Іншими словами, функції f_1, \dots, f_i повинні володіти наступної властивістю. Для всіх n -розрядних наборів, що містять будь-яке поєднання з не менше ніж $k + 1$ нульових розрядів, знайдеться r ($r < L$) функцій з множини (f_1, \dots, f_i) , які приймають нульові значення на цих наборах і позначають r таких ребер, видалення яких з графа G призводить до втрати його зв'язності.

Очевидно, що найбільш складними етапами при побудові GL - моделі є вибір графа G і функцій f_1, \dots, f_i . Для k - ВБС, використовуються графи кільцевого типу. Кільцевий граф спочатку стійкий до однієї відмови завдяки своїй структурі. Кільцевий граф й приймемо за базовий.

Використання додаткових ребер, безсумнівно, вирішує завдання трансформації GL -моделі базової ВБС в GL -модель небазової, однак, такий підхід в будь-якому випадку ускладнює граф моделі, ускладнюючи визначення його зв'язності, і тим самим збільшуючи витрати часу на розрахунок надійності системи. Крім того, завдання проведення додаткових ребер ускладнюється при наявності ситуації, коли деякі вектори стану заданої множини утворюють так звані «попарні реберні цикли».

Критичними будемо називати комбінації, число одиниць (відмовивших елементів) в якій дорівнює $k+1$.

Покриту комбінацію будемо називати таку критичну комбінацію, яка виходить при перемножуванні вже сформованих ребёрних функцій.

2.3 Висновки

У розділі розглядаються питання важливості побудови багатопроцесорних систем і їх місця в науці, техніці і в народному господарстві в цілому. Аналізуються різні підходи до вирішення завдань моделювання систем, в тому числі з використанням теорії графів. Наводяться міркування важливості побудови багатопроцесорних систем з наданням їм властивості відмовостійкості з метою підвищення їх надійності, в першу чергу для систем управління складними і відповідальними об'єктами. Відзначається, що при цьому зростає складність розрахунку надійності характеристик таких систем. Пропонується для розрахунку надійності відмовостійких систем використовувати графо-логічні моделі, описуються їх основні властивості, наводяться визначення базових та небазових графо-логічних моделей, їх особливості та можливості використання для розрахунку надійності складних відмовостійких багатопроцесорних систем управління статистичними методами. Відзначаються відмінні риси графо-логічних моделей: приписування булевих функцій ребрам графа. Функції залежать від змінних - станів відповідних процесорів.

3. ПОБУДОВА БАЗОВИХ GL-МОДЕЛЕЙ

3.1 Реберні функції базової GL-моделі і їх формування

3.1.1 2-відмовостійкі ВБС

При проектуванні відмовостійких багатопроцесорних систем необхідно вирішувати цілий ряд задач, в тому числі передбачити надлишкові програмно-апаратні засоби для тестування і виявлення відмовили модулів, а потім виконання процедур реконфігурації системи, що полягають в інформаційній ізоляції відмовили модулів, збереженні даних, використанні резерву (як структурного, так і тимчасового) і відновлення обчислювального процесу.

Очевидно, що для оптимального вибору апаратних і алгоритмічних ресурсів, що забезпечують реконфігурацію системи, необхідна досить точна оцінка параметрів надійності ВБС на ранніх етапах її проектування. У той же час відомі методи розрахунку параметрів надійності в разі ВБС дають неточну оцінку. Спроби наблизити розрахункові оцінки до реальних стикаються з непереборними обчислювальними труднощами, пов'язаними з величезним перебором можливих станів системи.

Сказане вище стосується, наприклад, до розрахунків надійності ВБС, заснованим на використанні моделі двополюсного графа. Критерієм працездатності системи в такій моделі є існування шляху між двома виділеними полюсами графа. Аналіз проводиться за умови випадкового існування ребер графа. При великій складності аналізованої системи завдання як побудови самого графа, так і перебору всіх можливих шляхів в двополюсній графі переходить в клас неконструктивних завдань.

У деяких випадках (наприклад, для мережевих структур) в якості моделі для розрахунку надійності може бути обраний багатополіусний

граф, і критерієм надійності є зв'язність графа (система вважається працездатною тільки тоді, коли кожен вузол графа має зв'язок з будь-яким іншим вузлом). Завдання полягає у визначенні ймовірності події, що складається в тому, що, по крайній мере, одне дерево ненаправленого графа працездатний. При цьому комбінаторна задача перебору всіх дерев для дуже складних мереж призводить, як і в випадку з двополюсними графами, до значних обчислювальних труднощів.

Методи статистичного моделювання, що дозволяють обійти проблему перебору дуже великого числа станів досліджуваної системи, за рахунок отримання статистичних оцінок з заданою вірогідністю, в значній мірі втрачають свою перевагу з - за, як правило, надзвичайно складного опису використовуваних критеріїв, в якості яких застосовуються структурні функції, характеризують працездатність системи.

Наведені приклади підтверджують актуальність завдання побудови нових моделей поведінки ВБС, які дозволили б зробити аналіз станів системи більш доступним і доступним для огляду, і в результаті спростити розрахунки і отримати досить точну оцінку параметрів надійності.

Відзначимо, що для ефективного аналізу поведінки систем, що мають велике число станів, вельми важливим є компактне подання критерію, відповідно до якого здійснюється оцінка поведінки системи в певних умовах.

Основна ідея нового підходу полягає в побудові моделі, що поєднує використання властивостей графів і булевих функцій. Таку модель в подальшому будемо називати графо-логічної. Нова модель передбачає використання для відображення поведінки системи відповідності між зв'язністю деякого графа і обчислюються для кожного стану системи значеннями певним чином обраних булевих функцій. Однак на відміну від відомих моделей, що використовують графи, в даному випадку використовується граф може мати набагато більш просту структуру, ніж

структура моделюється. Цього вдається досягти за рахунок «перенесення» складності на булеві функції, якими позначаються ребра графа. При цьому критерій, який використовується для визначення стану системи, може бути зроблений досить простим, якщо вдається вибрати регулярну форму подання булевих функцій в графо - логічній моделі.

Опишемо коротко основні принципи побудови графо - логічної моделі. Відзначимо, що головна ідея полягає в створенні можливості змінювати структуру графа відповідно до вектору стану елементів системи, що досягається шляхом приписування ребрах графа певних булевих функцій.

Спочатку зупинимося на більш простому випадку, коли система містить n -модулів і стає непрацездатною при відмові не менше $k+1$ модулів, за умови $k < n$. Таку ВБС будемо називати k -ВБС.

Поставимо у відповідність k -ВБС деякої зв'язної неорієнтований граф G , ребра якого позначаються певним чином обраними булевыми функціями $f_1(x_1, \dots, x_n), \dots, f_L(x_1, \dots, x_n)$, де кожна змінна x_i ($i = \overline{1, n}$) являється індикаторною функцією, яка позначає стан i -го елемента системи, тобто $x_i=1$, якщо i -й елемент системи працездатний, та $x_i=0$ в іншому випадку. Ребро, позначене функцією $f_i(x_1, \dots, x_n)$, залишається в графі G , якщо $f_i(x_1, \dots, x_n) = 1$ та удаляється з нього, якщо $f_i(x_1, \dots, x_n) = 0$.

Якщо вибрати функції $f_1(x_1, \dots, x_n), \dots, f_L(x_1, \dots, x_n)$ таким чином, щоб граф G втрачав зв'язність тільки при рівності нулю не менше $k+1$ будь-яких з n змінних, то відмова n -модульної системи можна інтерпретувати, як втрату зв'язності графа G . Іншими словами функції f_1, \dots, f_L повинні володіти наступною властивістю: для всіх n -розрядних наборів, що містять будь-яке поєднання з не менше, як $k+1$ нульових розрядів, знайдеться r ($r < L$) функцій з множини (f_1, \dots, f_L) , які приймають нульові значення на цих

наборах та позначають r таких ребер, видалення якого з графа G призводить до втрати його зв'язності.

Очевидно, що дуже важливими етапами при побудові графо - логічної моделі є вибір графа G і функцій f_1, \dots, f_L .

Найпростішим графом є незамкнутий ланцюг, а найпростішими функціями - $f_i = x_i$. Такий граф з ребрами, що помічені функціями $f_i = x_i$, може бути використан в графо – логічної моделі для 0 – ВБС. Якщо ланцюг замкнути, тобто отримати циклічний граф, то разом з цими ж функціями $f_i = x_i$ отриманий граф стане графо - логічною моделлю 1-ВБС.

Зупинимося далі на випадку, коли система містить n модулів і вона стає непрацездатною лише тоді, коли відмовляють не менше трьох модулів. За нашою термінологією така система є 2-ВБС. В якості графа G вибираємо неорієнтовний циклічний n -реберний граф (рис. 3.1), i -му ребру якого відповідає булева функція

$$f_i(x_1, \dots, x_n) = \prod_{j=1, \left\lfloor \frac{n-1}{2} \right\rfloor} (x_i \vee x_{(i+j) \bmod(n)}), \quad (3.1)$$

для $i = \overline{1, n}$.

Нехай також i -е ребро існує, якщо $f_i=1$, та відсутнє, якщо $f_i=0$.

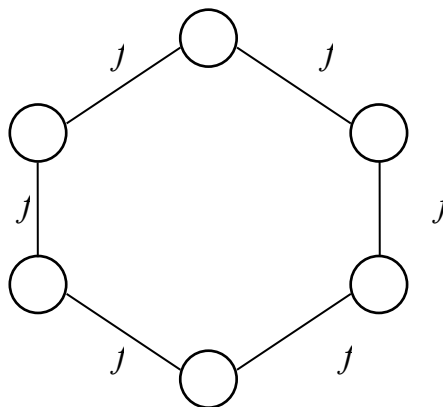


Рис. 3.1

Доведемо наступне.

Граф G втрачає зв'язність, якщо і тільки якщо не менше трьох змінних з безлічі $\{x_1, \dots, x_n\}$ приймають нульові значення.

Доведення.

Спочатку покажемо, що при рівності нулю менше трьох змінних, нульове значення може прийняти тільки одна функція.

Очевидно, що при рівності нулю однієї змінної $f_i \neq 0$ для будь-якого $i = \overline{1, n}$.

Нехай $x_p, x_q \in \{x_1, \dots, x_n\}$ і тільки вони рівні нулю. Відзначимо як очевидне, що в цьому випадку з усіх $f_i, i = \overline{1, n}$, тільки f_p та f_q можуть бути рівними нулю. Очевидно також, що $f_p = 0$, якщо виконується

$$q \in \left\{ (p+1) \bmod(n), (p+2) \bmod(n), \dots, \left(p + \left\lceil \frac{n-1}{2} \right\rceil \right) \bmod(n) \right\}, \quad (3.2)$$

а $f_q = 0$, якщо виконується

$$p \in \left\{ (q+1) \bmod(n), (q+2) \bmod(n), \dots, \left(q + \left\lceil \frac{n-1}{2} \right\rceil \right) \bmod(n) \right\}. \quad (3.3)$$

Легко показати, що одночасне виконання (3.2) та (3.3) при $p \neq q$ неможливе. Отже, висловлене припущення справедливо.

Покажемо тепер, що для трьох і більше рівних нулю змінних з $\{x_1, \dots, x_n\}$ принаймні дві функції f_k та f_l ($k = \overline{1, n}, l = \overline{1, n}, k \neq l$) приймають нульові значення.

Нехай $x_p, x_q, x_r \in \{x_1, \dots, x_n\}$ и $p < q < r$. Якщо виконується (2), то $f_p = 0$. При цьому або виконується

$$r \in \left\{ (q+1) \bmod(n), (q+2) \bmod(n), \dots, \left(q + \left\lceil \frac{n-1}{2} \right\rceil \right) \bmod(n) \right\}, \quad (3.4)$$

и тоді $f_q = 0$, або (3.4) не виконується і тоді, враховуючи, що $r + \left\lceil \frac{n-1}{2} \right\rceil > n$

, легко помітити, що

$$p \in \left\{ (r+1) \bmod(n), (r+2) \bmod(n), \dots, \left(r + \left\lfloor \frac{n-1}{2} \right\rfloor \right) \bmod(n) \right\}. \quad (3.5)$$

Отже, $f_r=0$.

Якщо умова (3.2) не виконується (при цьому, очевидно, що

$$r \notin \left\{ (p+1) \bmod(n), \dots, \left(p + \left\lfloor \frac{n-1}{2} \right\rfloor \right) \bmod(n) \right\}, \text{ то } f_p \neq 0. \text{ Але в цьому випадку}$$

очевидно виконання умов (4) і (5). Отже, $f_q=0$ и $f_r=0$.

Положення доведено.

Доведена теорема дозволяє легко будувати графо-логічні моделі для будь-яких значень n при $m=2$. Наприклад для випадку $n=6$, вона має вигляд, представлений на малюнку 2.1, де функції

$$\begin{aligned} f_1 &= x_1 \vee x_2 \cdot x_3 \\ f_2 &= x_2 \vee x_3 \cdot x_4 \\ f_3 &= x_3 \vee x_4 \cdot x_5 \\ f_4 &= x_4 \vee x_5 \cdot x_6 \\ f_5 &= x_5 \vee x_6 \cdot x_1 \\ f_6 &= x_6 \vee x_1 \cdot x_2 \end{aligned}$$

Ми бачимо, що функція f_{i+1} виходить з функції f_i шляхом додавання 1 по $\bmod n$ к індексам всіх змінних. Число ребер в точності рівно n .

Простота формування моделі робить її дуже привабливою. Однак спроба поширити цей спосіб формування для випадків $m>2$ не дає позитивного результату, тобто пряме узагальнення не має місця. У той же час деякі приватні результати для $m=3$ можна навести як приклади, що ілюструють неоднозначність рішень.

$$n = 6 \quad f_1 = x_1 \vee x_2 \vee x_5 \cdot x_6$$

$$n = 7 \quad f_1 = x_1 \vee x_2 \cdot x_3 \vee x_5 \cdot x_7$$

$$n = 9 \quad f_1 = x_1 \vee x_2 \cdot x_3 \vee x_6 \cdot x_7 \cdot x_8 \cdot x_9$$

$$n = 10 \quad f_1 = x_1 \vee x_4 \cdot x_7 \cdot x_{10} \vee x_2 \cdot x_5 \cdot x_8$$

$$n = 12 \quad f_1 = x_1 \vee x_2 \cdot x_3 \cdot x_4 \vee (x_2 \vee x_7) \cdot x_{10} \cdot x_{11} \cdot x_{12} \cdot x_8 \cdot x_9$$

$$n = 14 \quad f_1 = x_1 \vee x_2 \cdot x_3 \cdot x_4 \cdot x_5 \cdot (x_6 \vee x_7) \vee (x_8 \vee \overline{x_4} \vee \overline{x_5}) \cdot \\ \cdot (x_9 \vee \overline{x_5}) \cdot x_{14} \cdot x_{10} \cdot x_{11} \cdot x_{12} \cdot x_{13}$$

Функції f_{i+1} формуються з f_i точно так же, як вже описано вище.

Наведемо без доведення спільне рішення для 3-ВБС. Для простоти дамо вираз для випадку $n = 120$, з якого буде зрозумілий спосіб формування функцій для будь-якого n :

$$f_1 = x_1 \vee x_2 \cdot x_3 \cdot x_4 \cdot \dots \cdot x_{40} \cdot (x_{41} \vee x_{42}) \vee (\overline{x_{40}} \vee x_{79} \cdot (\overline{x_{39}} \vee x_{78} \cdot (\overline{x_{38}} \vee x_{77} \cdot (\overline{x_{37}} \vee x_{76} \cdot (\dots \\ \dots x_{44} \cdot (\overline{x_4} \vee x_{43})))))) \cdot x_{80} \cdot x_{81} \cdot x_{82} \cdot \dots \cdot x_{120}$$

Число функцій як і раніше дорівнює n , f_{i+1} формується з f_i так само просто, однак наведене рішення не можна визнати досить хорошим для практичних цілей хоча б тому, що функції не виходять неповторним. Крім того, залишається відкритим питання побудови моделей для інших значень величини m .

Це призводить до необхідності пошуку більш загального підходу до вирішення поставленого на початку параграфа завдання, нехай навіть шляхом ускладнення графа і його ребрових функцій.

3.1.2 GL-моделі загального виду

У попередньому параграфі відзначалися, з одного боку, необхідність відшукування загального рішення для довільних значень числа модулів і числа захищених відмов і, з іншого боку, неможливість прямого узагальнення отриманих там результатів. Тому в цьому параграфі займемося вирішенням поставленого завдання за допомогою іншого підходу.

Розглянемо k -ВБС (т.е. ВБС, захищену від відмов кратності не вище k), що складається з n елементів (модулей). Стан системи (відмовна ситуація) визначається поєднанням m відмовивших та $n-m$ справних елементів. Кожному стану системи можна поставити у відповідність деяку неупорядковану m - вибірку з n елементів. Надалі будемо говорити просто m -вибірка, маючи на увазі, що це неупорядкована вибірка.

Нехай $n_u = \{\alpha_1, \alpha_2, \dots, \alpha_{n_u}\}$ - деяку підмножину множини всіх елементів k -ВБС. Для k -ВБС визначимо безліч $K(m_v, n_u)$ як безліч, що складається з усіх m_v вибірок з n_u елементів.

Нехай $K(s, n_p) = \{a_1, a_2, \dots, a_g\}$, де a_i - деяка S -вибірка з n_p елементів ($i = \overline{1, g}$), а $K(t, n_r) = \{b_1, b_2, \dots, b_h\}$, де b_j - деяка t -вибірка з n_r елементів. Добуток множин $K(s, n_p)$ та $K(t, n_r)$ визначимо відповідно наступним чином:

$$K(s, n_p) \times K(t, n_r) = \{(a_1 \times b_1), (a_1 \times b_2), \dots, (a_1 \times b_h), \dots, (a_g \times b_1), (a_g \times b_2), \dots, (a_g \times b_h)\}$$

де $(a_i \times b_j)$ - результат конкатенації елементів вибірок a_i та b_j . Далі розіб'ємо n -множина, що представляє безліч всіх елементів системи, на q непересічних підмножин таким чином, що в 1-му підмножині буде n_1 елементів, у 2-му - n_2 елементів і т.д., в q -му - n_q елементів, тобто $n = \sum_{i=1}^q n_i$.

Тоді за аналогією з відомим комбінаторним тотожністю можна записати наступне рівність

$$\sum_{m_1, \dots, m_q} C_{n_1}^{m_1} \cdot C_{n_2}^{m_2} \cdot \dots \cdot C_{n_q}^{m_q} = C_n^m$$

(3.6)

$$K(m, n) = \bigcup_{m_1, m_2, \dots, m_q} K(m_1, n_1) \times K(m_2, n_2) \times \dots \times K(m_q, n_q),$$

(3.7)

де підсумовування в виразі (3.6) і операція об'єднання множин в виразі (3.7) виконуються для всіх таких наборів чисел m_1, m_2, \dots, m_q , для яких має місце $m = \sum_{i=1}^q m_i$ и $0 \leq m_i \leq n_i$.

Далі, використовуючи розбиття множини, представлене співвідношенням (3.7), займемося побудовою GL -моделі k -ВБС на основі графа кільцевого типу для довільних значень k . З огляду на, що граф кільцевого типу втрачає зв'язність тільки тоді, коли видаляється не менше двох ребер, легко встановлюється критерій відмови k -ВБС в термінах безлічі булевих функцій, якими позначені ребра цього графа. Таким критерієм є рівність нулю значень не менше, ніж двох функцій. Для того, щоб визначити процедуру вибору таких функцій, попередньо доведемо наступну теорему.

Нехай m -ВБС має n елементів, і множену, що складається з цих елементів, розбито на q підмножин з числом елементів n_1, n_2, \dots, n_q відповідно. Нехай також існує $f_1(x_1, \dots, x_{n_1}), \dots, f_L(x_1, \dots, x_{n_1})$ - такі булеві функції, що кожна з функцій $f_j (j \in \overline{1, L})$ приймає нульове значення, якщо дорівнюють нулю, по крайній мере, m змінних, що визначають стану m_i елементів для кожного з n_i - підмножин ($i \in \overline{1, q}$) відповідно до значень чисел m_1, m_2, \dots, m_q з i -го набору. Тоді справедливо наступне:

не менше двох функцій з безлічі f_1, \dots, f_L візьмуть нульові значення, якщо, принаймні, будь-які $m+1$ з n змінних дорівнюють нулю.

Спочатку покажемо, що при рівності нулю менш, ніж $m+1$ змінних, нульове значення може прийняти тільки одна функція з f_1, \dots, f_L . Для цього достатньо розглянути випадок, коли рівно m змінних дорівнюють нулю. З огляду на, що будь-яка m -вибірка з n змінних відповідає тільки одному набору значень m_1, m_2, \dots, m_q , висловлене припущення стає очевидним.

Перейдемо тепер до випадку, коли $m+1$ змінних приймають нульове значення. Нехай це будуть наступні змінні:

$$x_{\alpha_1} = 0, x_{\alpha_2} = 0, \dots, x_{\alpha_m} = 0, x_{\alpha_{m+1}} = 0, \text{ где числа (індекси) } (\alpha_1, \alpha_2, \dots, \alpha_m, \alpha_{m+1})$$

представляють довільну $(m+1)$ – вибірку індексів з n можливих.

З елементів цієї $(m+1)$ – вибірки можливо скласти

$$C_{m+1}^m = \frac{(m+1)!}{m!(m+1-m)!} = m+1 \text{ різних } m - \text{ виборок. Розглянемо наступні три з}$$

них: $(\alpha_1, \alpha_2, \dots, \alpha_{m-1}, \alpha_m)$, $(\alpha_1, \alpha_2, \dots, \alpha_{m-1}, \alpha_{m+1})$ и $(\alpha_2, \dots, \alpha_{m-1}, \alpha_m, \alpha_{m+1})$.

Далі для простоти будемо розуміти під вибіркою індексів $(\alpha_1, \alpha_2, \dots)$ вибірку змінних $(x_{\alpha_1}, x_{\alpha_2}, \dots)$ і відповідно під елементом α_i -елемент x_{α_i} .

Нехай вибірка $(\alpha_1, \alpha_2, \dots, \alpha_m)$ відповідає деякому набору (причому, цей набір єдиний) значень m_1, m_2, \dots, m_q з номером j . Покажемо, що хоча б одна з вибірок $\alpha_1, \alpha_2, \dots, \alpha_{m-1}, \alpha_{m+1}$ и $\alpha_2, \dots, \alpha_{m-1}, \alpha_m, \alpha_{m+1}$ не може відповідати набору з тим же номером j , який був встановлений для вибірки $\alpha_1, \alpha_2, \dots, \alpha_m$. Дійсно, якщо зі структури вибірки $\alpha_1, \alpha_2, \dots, \alpha_{m-1}, \alpha_{m+1}$ слідує, що α_{m+1} не входить у підмножина p_i , що містить елемент α_m , то очевидно відбувається зміна набору m_1, m_2, \dots, m_q . У будь-якому випадку знаходиться α_{m+1} в одному підмножині разом з одним з елементів $\alpha_1, \alpha_2, \dots, \alpha_{m-1}$; чи ні, - відбудеться збільшення на одиницю значення деякого розряду набору m_1, m_2, \dots, m_q . Якщо ж виявляється, що α_{m+1} знаходиться в одному підмножині p_i разом з елементом α_m , то вибірці $(\alpha_1, \alpha_2, \dots, \alpha_{m-1}, \alpha_{m+1})$ відповідає той же набір з номером j , що і вибірці $(\alpha_1, \alpha_2, \dots, \alpha_m)$. Але при цьому третя вибірка $(\alpha_2, \dots, \alpha_{m-1}, \alpha_m, \alpha_{m+1})$ дає зміна набору, так як значення розряду m_i набору m_1, m_2, \dots, m_q збільшується на одиницю.

Таким чином, три розглянутих вибірки повинні відповідати не менше, ніж двом різним наборам значень чисел m_1, m_2, \dots, m_q , а це означає,

що серед функцій f_1, \dots, f_L знайдуться, щонайменше, дві функції, які візьмуть нульові значення, що і потрібно було показати.

Покажемо тепер, що будь-яка функція f_i з розглянутого безлічі функцій $f_1(x_1, \dots, x_n), \dots, f_L(x_1, \dots, x_n)$, задовольняють наведеним вище умовам, може бути представлена в наступному вигляді:

$$f_i = f_{m_1} \vee f_{m_2} \vee \dots \vee f_{m_q}, \quad (3.8)$$

де,

$$f_{m_j} = y_{a_1} \wedge y_{a_2} \wedge \dots \wedge y_{a_{N_j}}; j = \overline{1, q}; N_j = C_{n_j}^{m_j}$$

$$y_{a_t} = x_{\alpha_1} \vee x_{\alpha_2} \vee \dots \vee x_{\alpha_{m_j}}; t = \overline{1, C_{n_j}^{m_j}}.$$

Відзначимо також, що функції f_{m_j} - це функції, які беруть нульові значення тільки тоді, коли дорівнюють нулю всі змінні хоча б з однією m_j - вибірки з n_j - підмножини.

Дійсно, нехай $(x_{\alpha_1}, x_{\alpha_2}, \dots, x_{\alpha_m})$ - деякий набір m змінних з n – безлічі, і кожна змінна в цьому наборі набуває нульових значень. Очевидно, що в цьому наборі присутня група змінних, що представляє якусь m_1 -вибірку з n_1 – підмножини, група змінних, що представляє якусь m_2 -вибірку з n_2 – підмножини, і так далі до групи змінних, що представляють яку-небудь m_q -вибірку из n_q – підмножини. Групи змінних, що становлять цей набір, відповідають деякого єдиному набору чисел (m_1, m_2, \dots, m_q) .

$$y_{a_1} = x_{\alpha_1} \vee x_{\alpha_2} \vee \dots \vee x_{\alpha_{m_1}} = 0,$$

$$f_{m_1} = y_{a_1} \wedge y_{a_2} \wedge \dots \wedge y_{a_{N_1}} = 0, (N_1 = C_{n_1}^{m_1}).$$

Нехай, наприклад, m_1 -вибірка представляє набір змінних $(x_{\alpha_1}, x_{\alpha_2}, \dots, x_{\alpha_{m_1}})$.

Аналогічно, розглядаючи набори змінних, складових m_2 - вибірку і т.д. m_q - вибірку, отримаємо, що $f_{m_2}=0, \dots, f_{m_q}=0$. Отже, в цьому випадку функція f_i з виразу (3.8) приймає нульове значення.

Розглянемо тепер набір змінних, групи змінних, якого відповідають деякому набору чисел $\mathbf{m}'=(m_1', m_2', \dots, m_q')$, відмінному від розглянутого вище набору $\mathbf{m}=(m_1, m_2, \dots, m_q)$. Якщо набори \mathbf{m} та \mathbf{m}' відрізняються, то для того, щоб виконувалися рівності

$$\sum_{i=1}^q m_i = m,$$

$$\sum_{i=1}^q m_i' = m,$$

необхідно щоб виконувалися також два наступних нерівностей: $m_p' < m_p$ и $m_r' > m_r$, де p та r ($p \neq r$) деякі цілі числа, що задовольняють нерівностям $1 \leq p \leq q$ и $1 \leq r \leq q$. Звідси випливає, що група змінних, складова будь-яку m_r' - вибірку з n_r – безлічі містить хоча б одну змінну, значення якої дорівнює одиниці. При цьому отримаємо:

$$y_{a_1} = 1, y_{a_2} = 1, \dots, y_{a_{N_j}} = 1; f_{m_j} = 1; f_i = 1.$$

Нарешті, якщо число змінних в даному наборі менше числа m , то це означає, що в одному з підмножин, наприклад в n_z – підмножині будуть змінні, що мають одиничні значення. відповідно отримаємо $f_i = 1$.

Таким чином, функція f_i приймає нульове значення тільки в тому випадку, якщо не менше m змінних приймають нульові значення. Причому, ці змінні повинні представляти m_1 -вибірки, m_2 -вибірки, \dots , m_q -вибірки з n_1 -підмножини, n_2 -підмножини, \dots , n_q -підмножини відповідно, для єдиного набору чисел (m_1, m_2, \dots, m_q) .

При побудові GL-моделі, яка описує поведінку k-ВБС, запис булевих функцій $f_1(x_1, \dots, x_n), \dots, f_L(x_1, \dots, x_n)$, що відповідають сформульованим вище умов, зручно виконати за таким правилом:

1. Провести розбиття множини з n – елементів, що становлять k -ВБС, на q непересічних підмножин з числом елементів відповідно n_1, n_2, \dots, n_q

$$(\sum_{i=1}^q n_i = n).$$

2. Перерахувати всілякі набори значень m_1, m_2, \dots, m_q за умови, що $\sum_{i=1}^q m_i = m$, $m_i \leq n_i$.

3. Для кожного з наборів значень чисел m_1, m_2, \dots, m_q , записати вираз виду

$$\begin{aligned} K(m_1, n_1) \times K(m_2, n_2) \times \dots \times K(m_q, n_q) = \\ = \{(a_1 \times b_1 \times \dots \times c_1), (a_1 \times b_1 \times \dots \times c_2), \dots, (a_{n_1} \times b_{n_2} \times \dots \times c_{n_q})\}, \end{aligned}$$

де a_i, b_j, \dots, c_k – це m_1 – вибірки з n_1 – множини, m_2 – вибірки з n_2 – множини, \dots, m_q – вибірки з n_q – множини ($i = \overline{1, C_{n_1}^{m_1}}, j = \overline{1, C_{n_2}^{m_2}}, \dots, k = \overline{1, C_{n_q}^{m_q}}$), тобто a_i вибірка виду $\{\alpha_1, \alpha_2, \dots, \alpha_{m_1}\}$, b_j вибірка виду $\{\beta_1, \beta_2, \dots, \beta_{m_2}\}$, \dots, c_k вибірка виду $\{\gamma_1, \gamma_2, \dots, \gamma_{m_q}\}$.

4. Здійснити перехід від вираження з п.3 до булевих виразів. булева функція $f_i(x_1, \dots, x_n)$, де i – номер набору значень чисел m_1, m_2, \dots, m_q ($i = \overline{1, L}$), записується у вигляді:

$$f_i = (\bigwedge_i y_{a_i}) \vee (\bigwedge_j y_{b_j}) \vee \dots \vee (\bigwedge_k y_{c_k})$$

$$\text{где } y_{a_i} = \bigvee_{\sigma=1}^{m_1} x_{\alpha_\sigma}, y_{b_j} = \bigvee_{\theta=1}^{m_2} x_{\beta_\theta}, \dots, y_{c_k} = \bigvee_{\omega=1}^{m_q} x_{\gamma_\omega}$$

Розглянемо варіант розбиття n – множини, при якому в певній мірі спрощується запис функцій f_1, \dots, f_L . Такий варіант розбиття буде досягнутий, коли в вираженні (3.6) для всіх підмножин $K(m_j, n_j)$ має місце наступне: або $m_j = 1$, або $m_j = n_j$. Якщо $m_j = 1$, отримаємо в виразах (3.8) $y_{a_i} = x_{a_i}, t = \overline{1, n_j}$ і, отже, вирази для f_{mj} представляють елементарні кон'юнкції $f_{m_j} = x_{\alpha_1} \wedge x_{\alpha_2} \wedge \dots \wedge x_{\alpha_{n_j}}$. Якщо $m_j = n_j$, отримаємо в виразах (3.8) $y_{a_i} = x_{a_i}, t = \overline{1, n_j}$ і, отже, вирази для f_{mj} представляють елементарні диз'юнкції $f_{m_j} = x_{\alpha_1} \vee x_{\alpha_2} \vee \dots \vee x_{\alpha_{n_j}}$. Відповідно при виборі розглянутого варіанту розбиття n -безлічі отримуємо функції f_i у виразах (3.8) з більш простою формою запису. Тому розбиття n -множини доцільно здійснювати в кілька етапів. На першому етапі отримуємо n_1, n_2, \dots, n_q підмножини. Після першого етапу розбиття подальше розбиття здійснюємо тільки для таких підмножин n_i , вираз яких не відповідає $K(1, n_j)$ або $K(n_j, n_j)$.

3.1.3 Приклад

Нехай дана система є 3-ВБС, і складається з 9 елементів, позначених номерами 1, 2, 3, 4, 5, 6, 7, 8, 9. Необхідно побудувати GL-модель для даної системи, використовуючи граф кільцевого типу. Розіб'ємо безліч елементів системи на 3 підмножини: $n_1 = \{1, 2, 3\}$, $n_2 = \{4, 5, 6\}$ и $n_3 = \{7, 8, 9\}$. Відповідно до виразом (3.6), отримаємо для безлічі $K(3, 9)$:

$$K(m, n) = \bigcup_{m_1, m_2, m_3} K(m_1, n_1) \times K(m_2, n_2) \times K(m_3, n_3),$$

де операція об'єднання проводиться для всіх наборів чисел m_1, m_2, m_3 , причому $\sum_{i=1}^3 m_i = 3$, $0 \leq m_i \leq 3$. Для визначення множин $K(m_i, n_i)$ складемо таблицю всіх можливих наборів чисел m_1, m_2, m_3 для умов даного прикладу.

Таблиця 2.1

	Значення чисел
--	----------------

№ набору	m_1	m_2	m_3
1	0	0	3
2	0	1	2
3	0	2	1
4	0	3	0
5	1	0	2
6	1	1	1
7	1	2	0
8	2	0	1
9	2	1	0
10	3	0	0

Тепер вираз для $K(3,9)$ можна записати в такий спосіб:

$$K(3,9) = (K(0,n_1) \times K(0,n_2) \times K(3,n_3)) \cup (K(0,n_1) \times K(1,n_2) \times K(2,n_3)) \cup \\ \cup (K(0,n_1) \times K(2,n_2) \times K(1,n_3)) \cup \dots \cup (K(3,n_1) \times K(0,n_2) \times K(0,n_3))$$

.

В цей вислів входять порожні безлічі $K(0,n_i)$ для всіх $i = \overline{1,3}$, а

також

множини

$$K(3,n_3) = \{(789)\}; K(1,n_2) = \{4,5,6\}; K(2,n_3) = \{(78), (79), (89)\}; \\ K(2,n_2) = \{(45), (46), (56)\}; K(1,n_3) = \{(7), (8), (9)\}; \dots; K(3,n_1) = \{(123)\}$$

.

З таблиці 2.1 і з виразів для множин $K(m_j, n_j)$ видно, що не всі ці безлічі, отримані після розбивки n -множини, являють собою множини типу $K(n_j, n_j)$ або $K(1, n_j)$. Так, для наборів чисел m_1, m_2, m_3 з номерами 2, 3,

5, 7, 8, 9 маємо безлічі виду $K(2,3)$. Проведемо подальші розбиття таких множин. Результат представлений в таблиці 2.2.

Таблиця 2.2

№ наборів	$n_1=\{1, 2, 3\}$		$n_2=\{4, 5, 6\}$		$n_3=\{7, 8, 9\}$	
	$n_{1_1} = \{1,2\}$	$n_{1_2} = \{3\}$	$n_{2_1} = \{4,5\}$	$n_{2_2} = \{6\}$	$n_{3_1} = \{7,8\}$	$n_{3_2} = \{9\}$
	m_1		m_2		m_3	
	m_{1_1}	m_{1_2}	m_{2_1}	m_{2_2}	m_{3_1}	m_{3_2}
1	0		0		3	
2	0		1		1	1
3	0		1		2	0
4	0		3		0	
5	1		0		1	1
6	1		0		2	0
7	1		1		1	
8	1		1	1	0	
9	1		2	0	0	
10	1	1	0		1	
11	2	0	0		1	
12	1	1	1		0	
13	2	0	1		0	
14	3		0		0	

Перейдемо тепер до отримання булевих рівнянь для GL-моделі. Використовуючи вищенаведене правило записи булевих функцій для GL-моделі, для кожного набору чисел m_1, m_2, m_3 і для кожного набору $m_{1_1}, m_{1_2}, m_{2_1}, m_{2_2}, m_{3_1}, m_{3_2}$ з таблиці 2.2 записуємо такі функції:

$$\begin{aligned}
f_1 &= x_7 \vee x_8 \vee x_9; \\
f_2 &= x_4 \cdot x_5 \cdot x_6 \vee x_7 \cdot x_8 \vee x_9; \\
f_3 &= x_4 \cdot x_5 \cdot x_6 \vee x_7 \vee x_8; \\
f_4 &= x_4 \vee x_5 \vee x_6; \\
f_5 &= x_1 \cdot x_2 \cdot x_3 \vee x_7 \cdot x_8 \vee x_9; \\
f_6 &= x_1 \cdot x_2 \cdot x_3 \vee x_7 \vee x_8; \\
f_7 &= x_1 \cdot x_2 \cdot x_3 \vee x_4 \cdot x_5 \cdot x_6 \vee x_7 \cdot x_8 \cdot x_9; \\
f_8 &= x_1 \cdot x_2 \cdot x_3 \vee x_4 \cdot x_5 \vee x_6; \\
f_9 &= x_1 \cdot x_2 \cdot x_3 \vee x_4 \vee x_5; \\
f_{10} &= x_1 \cdot x_2 \cdot x_3 \vee x_4 \cdot x_5 \cdot x_6 \vee x_7 \cdot x_8 \cdot x_9; \\
f_{11} &= x_1 \cdot x_2 \vee x_3 \vee x_7 \cdot x_8 \cdot x_9; \\
f_{12} &= x_1 \cdot x_2 \vee x_7 \cdot x_8 \cdot x_9; \\
f_{12} &= x_1 \cdot x_2 \vee x_3 \vee x_4 \cdot x_5 \cdot x_6; \\
f_{13} &= x_1 \vee x_2 \vee x_4 \cdot x_5 \cdot x_6; \\
f_{14} &= x_1 \vee x_2 \vee x_3;
\end{aligned}$$

Пропонована методика вирішує завдання побудови GL моделі для k -ВБС при довільних значеннях k . Особливістю методики є регулярний спосіб побудови булевих функцій, використовуваних для позначення ребер кільцевого графа, на основі якого будується дана модель.

Наведена теорема і наступна схема отримання булевих виразів можуть здатися складними і малопридатними для практичного використання. Однак насправді це не так, оскільки знайдені вирази для $K(m,n)$ можна використовувати при отриманні виразів $K(s,r)$, де $s \geq m$, а $r \geq n$. Покажемо це на прикладі формування GL- моделі для $K(4,8)$ двома способами.

1. Як вже зазначалося, найбільш простим буде формування булевих виразів для $K(1,i)$ або $K(i,i)$. Використовуючи цю властивість, можемо відразу записувати реберні функції, розбивши безліч змінних $\{x_1, x_2, \dots, x_8\}$ на 4 підмножини $\{x_1, x_2\}$, $\{x_3, x_4\}$, $\{x_5, x_6\}$ и $\{x_7, x_8\}$. Для кожного з підмножин можемо записати $K(1, 2) = x_i \cdot x_{i+1}$

$$K(2, 2) = x_i \vee x_{i+1}$$

Отже, перераховуємо сукупності $K(i, j)$, відповідні всіляким варіантам розподілів відмов по підмножини (див. табл. 2.3), і відповідні функції.

Таблиця. 2.3

$\{x_1, x_2\}$	$\{x_3, x_4\}$	$\{x_5, x_6\}$	$\{x_7, x_8\}$	
$K(1, 2)$	$K(1, 2)$	$K(1, 2)$	$K(1, 2)$	$f_1 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 x_8$
$K(2, 2)$		$K(1, 2)$	$K(1, 2)$	$f_2 = x_1 \vee x_2 \vee x_5 x_6 \vee x_7 x_8$
$K(2, 2)$	$K(1, 2)$		$K(1, 2)$	$f_3 = x_1 \vee x_2 \vee x_3 x_4 \vee x_7 x_8$
$K(2, 2)$	$K(1, 2)$	$K(1, 2)$		$f_4 = x_1 \vee x_2 \vee x_3 x_4 \vee x_5 x_6$
$K(1, 2)$	$K(2, 2)$	$K(1, 2)$		$f_5 = x_1 x_2 \vee x_3 \vee x_4 \vee x_5 x_6$
$K(1, 2)$	$K(2, 2)$		$K(1, 2)$	$f_6 = x_1 x_2 \vee x_3 \vee x_4 \vee x_7 x_8$
	$K(2, 2)$	$K(1, 2)$	$K(1, 2)$	$f_7 = x_3 \vee x_4 \vee x_5 x_6 \vee x_7 x_8$
$K(1, 2)$	$K(1, 2)$	$K(2, 2)$		$f_8 = x_1 x_2 \vee x_3 x_4 \vee x_5 \vee x_6$
$K(1, 2)$		$K(2, 2)$	$K(1, 2)$	$f_9 = x_1 x_2 \vee x_5 \vee x_6 \vee x_7 x_8$
	$K(1, 2)$	$K(2, 2)$	$K(1, 2)$	$f_{10} = x_3 x_4 \vee x_5 \vee x_6 \vee x_7 x_8$
$K(1, 2)$	$K(1, 2)$		$K(2, 2)$	$f_{11} = x_1 x_2 \vee x_3 x_4 \vee x_7 \vee x_8$
$K(1, 2)$		$K(1, 2)$	$K(2, 2)$	$f_{12} = x_1 x_2 \vee x_5 x_6 \vee x_7 \vee x_8$
	$K(1, 2)$	$K(1, 2)$	$K(2, 2)$	$f_{13} = x_3 x_4 \vee x_5 x_6 \vee x_7 \vee x_8$
$K(2, 2)$	$K(2, 2)$			$f_{14} = x_1 \vee x_2 \vee x_3 \vee x_4$

$K(2, 2)$		$K(2, 2)$		$f_{15} = x_1 \vee x_2 \vee x_5 \vee x_6$
$K(2, 2)$			$K(2, 2)$	$f_{16} = x_1 \vee x_2 \vee x_7 \vee x_8$
	$K(2, 2)$	$K(2, 2)$		$f_{17} = x_3 \vee x_4 \vee x_5 \vee x_6$
	$K(2, 2)$		$K(2, 2)$	$f_{18} = x_3 \vee x_4 \vee x_7 \vee x_8$
		$K(2, 2)$	$K(2, 2)$	$f_{19} = x_5 \vee x_6 \vee x_7 \vee x_8$

Вийшов 19-реберний кільцевий граф.

2. Можливі й інші варіанти. Зокрема, можна вихідне безліч змінних розбити на дві підмножини: $\{x_1, x_2, x_3, x_4\}$ та $\{x_5, x_6, x_7, x_8\}$. Комбінації різних $K(i, j)$, відповідні варіантам розподілу відмов по підмножини наведені в табл. 2.4.

Таблиця 2.4

$\{x_1, x_2, x_3, x_4\}$	$\{x_5, x_6, x_7, x_8\}$
$K(4, 4)$	
$K(3, 4)$	$K(1, 4)$
$K(2, 4)$	$K(2, 4)$
$K(1, 4)$	$K(3, 4)$
	$K(4, 4)$

Розіб'ємо кожне з підмножин змінних на свої підмножини: $\{x_1, x_2\}$, $\{x_3, x_4\}$, $\{x_5, x_6\}$ и $\{x_7, x_8\}$. Запишемо далі для кожного з $K(3, 4)$ та $K(2, 4)$ свої таблиці комбінацій $K(2, 2)$ та $K(1, 2)$ (див. табл. 2.5, 2.6, 2.7 та 2.8).

Таблиця 2.5

$\{x_1, x_2\}$	$\{x_3, x_4\}$
$K(2, 2)$	$K(1, 2)$
$K(1, 2)$	$K(2, 2)$

Таблиця 2.6

$\{x_1, x_2\}$	$\{x_3, x_4\}$
$K(2, 2)$	
$K(1, 2)$	$K(1, 2)$
	$K(2, 2)$

Таблиця 2.7

$\{x_5, x_6\}$	$\{x_7, x_8\}$
$K(2, 2)$	$K(1, 2)$
$K(1, 2)$	$K(2, 2)$

Таблиця 2.8

$\{x_5, x_6\}$	$\{x_7, x_8\}$
$K(2, 2)$	
$K(1, 2)$	$K(1, 2)$
	$K(2, 2)$

Тепер, коли все $K(i, j)$ задовольняють зазначеної вище (перший спосіб формування) особливості, неважко записати реберні функції GL-моделі (пам'ятаючи про конкатенації).

$$f_1 = x_1 \vee x_2 \vee x_3 \vee x_4$$

$$f_2 = x_1 \vee x_2 \vee x_3 x_4 \vee x_5 x_6 x_7 x_8$$

$$f_3 = x_1 x_2 \vee x_3 \vee x_4 \vee x_5 x_6 x_7 x_8$$

$$f_4 = x_1 \vee x_2 \vee x_3 \vee x_4$$

$$f_5 = x_1 x_2 \vee x_3 x_4 \vee x_5 \vee x_6$$

$$f_6 = x_3 \vee x_4 \vee x_5 \vee x_6$$

$$f_7 = x_1 \vee x_2 \vee x_5 x_6 \vee x_7 x_8$$

$$f_8 = x_1 x_2 \vee x_3 x_4 \vee x_5 x_6 \vee x_7 x_8$$

$$f_9 = x_3 \vee x_4 \vee x_5 x_6 \vee x_7 x_8$$

$$f_{10} = x_1 \vee x_2 \vee x_7 \vee x_8$$

$$f_{11} = x_1 x_2 \vee x_3 x_4 \vee x_7 \vee x_8$$

$$f_{12} = x_3 \vee x_4 \vee x_7 \vee x_8$$

$$f_{13} = x_1 x_2 x_3 x_4 \vee x_5 \vee x_6 \vee x_7 x_8$$

$$f_{14} = x_1 x_2 x_3 x_4 \vee x_5 x_6 \vee x_7 \vee x_8$$

$$f_{15} = x_5 \vee x_6 \vee x_7 \vee x_8$$

Як бачимо, в обох випадках число функцій (відповідно, ребер графа) більше числа модулів ВБС, проте функції не тільки неповторювані, а й іноді залежать не від усіх змінних, що полегшує роботу з ними.

На закінчення відзначимо, що дослідження так званих систем « k з n » присвячено досить багато робіт особливо в останні роки. *K-out-of n*: - система працездатна, якщо k модулів з n функціонують, тобто в нашому визначенні це - базова система. Цей інтерес зайвий раз підкреслює теоретичну і практичну важливість проблеми з одного боку і складність самої проблеми (а ВБС базові - найпростіші об'єкти) - з іншого. Розглядаються різні алгоритми оптимізації процесу визначення стану

системи, пропонуються формули розрахунку надійності. При цьому, як правило, передбачається наявність будь-яких обмежень, що полегшують рішення задачі.

3.2 Реберні функції на основі КНФ

Розглянемо k -ВБС (т.е. ВБС, захищену від відмов кратності не вище k), що складається з n елементів (модулів). Стан системи (отказові ситуація) визначається поєднанням m відмовивших і $n-m$ справних елементів. Кожному стану системи можна поставити у відповідність деяку неупорядковану m -вибірку з n елементів.

Безліч елементів ВБС розіб'ємо на 2 непересічних підмножини X_1 і X_2 з числом елементів в кожному n_1 і n_2 , відповідно. На основі відомого комбінаторного тотожності [3]

$$\sum_{k_1, k_2} C_{n_1}^{k_1} \cdot C_{n_2}^{k_2} = C_n^{k+1},$$

де підсумовування виконується для всіх наборів чисел k_1, k_2 , для яких мають місце умови $k_1 + k_2 = k + 1$, $0 \leq k_1 \leq n_1$ и $0 \leq k_2 \leq n_2$, розіб'ємо безліч всіх критичних комбінацій для даної системи на $k+2$ непересічних підмножини

$$\kappa_1 = K(0, n_1) \times K(k+1, n_2),$$

$$\kappa_2 = K(1, n_1) \times K(k, n_2),$$

$$\kappa_3 = K(2, n_1) \times K(k-1, n_2),$$

...

$$\kappa_{k+1} = K(k+1, n_1) \times K(0, n_2),$$

де $K(m_v, n_u)$ – множина, що складається з усіх m_v - вибірок з n_u елементів [2].

Шаблоном множини будемо називати структурний його представлення у вигляді визначення вибірок з підмножин цієї множини, об'єднання яких відповідає кожному елементу множини. Наприклад, шаблон $P(\{\{x_1, x_2, x_3\}, \{x_4, x_5, x_6, x_7\}\}, \{2, 3\})$ відповідає множини, в яку входять елементи: $\{x_1, x_2, x_4, x_5, x_6\}, \{x_1, x_3, x_4, x_5, x_7\}, \{x_2, x_3, x_4, x_5, x_7\}, \dots, \{x_2, x_3, x_5, x_6, x_7\}$.

Легко підрахувати, скільки критичних комбінацій включає в себе шаблон: $C_{n_1}^{k_1} \cdot C_{n_2}^{k_2} = C_3^2 \cdot C_4^3 = 3 \cdot 4 = 12$.

Кожному підмножині критичних комбінацій k_i поставимо у відповідність шаблон $P_i(\{X_1, X_2\}, \{i, k+1-i\})$. Для кожного шаблону визначимо функцію, що *покриває булеву функцію* f_i , яка має вигляд:

$$f_i = D_{n_1}^{k_1}(\{x_1, \dots, x_{n_1}\}) \cap D_{n_2}^{k_2}(\{x_{n_1+1}, \dots, x_{n_1+n_2}\}),$$

де $D_p^q(\{x_i, \dots, x_{i+p-1}\})$ – будемо позначати диз'юнкцію елементарних кон'юнкція змінних з підмножини $\{x_i, \dots, x_{i+p-1}\}$, які представляють собою всі q -вибірки з нього. Наприклад, для наведеного вище шаблону покриває функція буде виглядати так:

$$f = (x_1 x_2 \cup x_1 x_3 \cup x_2 x_3) \cap (x_4 x_5 x_6 \cup x_4 x_5 x_7 \cup x_4 x_6 x_7 \cup x_5 x_6 x_7)$$

Властивість покриває функції: при критичних комбінаціях, що належать відповідним їй шаблоном, завжди приймає одиничне значення. На інших, котрі належать до відповідного їй шаблоном критичних комбінаціях і при комбінаціях змінних з числом одиниць (відмов) меншим $k+1$ функція дорівнює завжди нулю. Наприклад, що покриває булева функція, наведена вище, буде приймати значення одиниці при критичних комбінаціях $x_1 = x_2 = x_4 = x_5 = x_6 = 1$, $x_2 = x_3 = x_5 = x_6 = x_7 = 1$ і т.п., при комбінації $x_1 = x_2 = x_3 = x_5 = x_6 = 1$ або $x_2 = x_3 = x_5 = x_6 = 1$ вона буде рівна 0.

Кільцевій граф, який лежить в основі GL-моделі, втрачає зв'язність при випаданні хоча б двох будь-яких його ребер, тобто при рівності одиниці двох і більше його ребрових функцій. Для визначення алгоритму побудови таких функцій розглянемо такий випадок.

Нехай k -ВБС має n елементів і їх множина розбита на два взаємно непересічних підмножини X_1 і X_2 з числом елементів n_1 і n_2 , відповідно ($n_1+n_2=n$). Нехай також існує безліч булевих функцій F , таке, що дві функції з нього мають вигляд:

$$\begin{aligned} f_1 &= D_{n_1}^i(X_1), \\ f_2 &= D_{n_2}^{k+1-i}(X_2), \end{aligned}$$

де i - довільне число в проміжку $\{1..k\}$, а решта представляють собою покривають функції для кожного шаблону

$$p_j(\{X_1, X_2\}, \{j, k+1-j\}),$$

де $j \neq i$ у проміжку $\{1..k\}$.

Справедливо наступне.

По крайній мере, дві функції з безлічі F візьмуть поодинокі значення тільки тоді, коли $k+1$ і більше змінних дорівнюють одиниці.

Доведення. Розглянемо випадок, коли $k+1$ змінних приймають середнє арифметичне значення. Нехай k'_1 з них належать підмножині змінних X_1 , а k'_2 - підмножині X_2 , враховуючи що $k'_1+k'_2=k+1$. Має місце наступна система нерівностей:

$$\left[\begin{aligned} &\begin{cases} k'_1 = i, \\ k'_2 = k+1-i. \end{cases} \quad (1) \\ &\begin{cases} k'_1 > i, \\ k'_2 < k+1-i. \end{cases} \quad (2) \\ &\begin{cases} k'_1 < i, \\ k'_2 > k+1-i. \end{cases} \quad (3) \end{aligned} \right.$$

де i - довільне ціле число з $[1..k+1]$, узятє при визначенні функцій f_1 і f_2 .

У разі, коли k'_1 і k'_2 задовольняють умовам системи (1), очевидно, що булеві функції f_1 і f_2 обидві візьмуть поодинокі значення, виходячи з визначення диз'юнктивній форми їх запису $D_{n_1}^i(X_1)$ і $D_{n_2}^{k+1-i}(X_2)$, відповідно.

У разі (2) середнє арифметичне значення прийме функція $f_1 = D_{n_1}^i(X_1)$ в зв'язку з тим, що знайдеться як мінімум дві елементарні кон'юнкції в її записи, які будуть дорівнюють одиниці в цьому випадку, наприклад, якщо

$$x_{a_1} = x_{a_2} = x_{a_3} = \dots x_{a_{i+1}} = 1 \quad \text{і} \quad x_{a_1}, x_{a_2}, \dots, x_{a_{i+1}} \in X_1, \quad \text{то} \quad \bigcap_{p=a_1}^{a_i} x_{a_p} = 1 \quad \text{і} \quad \bigcap_{p=a_2}^{a_{i+1}} x_p = 1.$$

Знайдеться також така булева функція $f_j \in F$, яка буде покриває для шаблону $p_j(\{X_1, X_2\}, \{k'_1, k'_2\})$ і, відповідно, буде дорівнювати одиниці в цьому випадку, з огляду на властивості покриває функції.

У разі (3) значення одиниці прийме функція $f_2 = D_{n_2}^{k+1-i}(X_2)$, в записі якої, як і в випадку (2), знайдеться як мінімум дві елементарні кон'юнкції рівні одиниці, а також покриває функція $f_j \in F$ шаблону $p_j(\{X_1, X_2\}, \{k'_1, k'_2\})$.

Тепер розглянемо варіант, коли число змінних з X рівних одиниці (відмовили елементів) менше довжини критичної комбінації $k+1$. Без обмеження спільності докази, припустимо кількість таких змінних одно k . Виходячи з визначення покриває функції шаблону, жодна з таких булевих функцій, що належать F ніколи не прийме середнє арифметичне значення при кількості змінних рівних одиниці меншому довжини критичної комбінації. Що стосується функцій f_1 і f_2 , то дорівнювати одиниці в цьому випадку може лише одна з них, з огляду на наступні умови:

$$\left[\begin{array}{l} \left\{ \begin{array}{l} k'_1 \geq i, \\ k'_2 < k+1-i. \end{array} \right. \Rightarrow \left\{ \begin{array}{l} f_1 = 1, \\ f_2 = 0. \end{array} \right. \\ \left\{ \begin{array}{l} k'_1 < i, \\ k'_2 \geq k+1-i. \end{array} \right. \Rightarrow \left\{ \begin{array}{l} f_1 = 0, \\ f_2 = 1. \end{array} \right. \\ \left\{ \begin{array}{l} k'_1 < i, \\ k'_2 < k+1-i. \end{array} \right. \Rightarrow f_1 = f_2 = 0 \end{array} \right.$$

Розглянуті вище випадки є вичерпними.

На основі доведеного положення можна визначити наступний алгоритм побудови безлічі ребрових функцій F для GL-моделі k -ВБС на кільцевому графі.

3.2.1 Алгоритм.

1) Розбити безліч X на дві підмножини (X_1 и X_2 , $|X_1|=n_1$, $|X_2|=n_2$), тобто визначити n_1 і n_2 .

2) Визначити функцій f_1 і f_2 відповідно до отриманих раніше n_1 і n_2 .

$$f_1 = D_{n_1}^i(X_1),$$
$$f_2 = D_{n_2}^{k+1-i}(X_2).$$

Величина i визначається з проміжку $\{1..k\}$ з урахуванням максимуму величини суми $(C_{n_1}^i + C_{n_2}^{k+1-i})$ з метою оптимізації.

3) Для кожного шаблону $p_j(\{X_1, X_2\}, \{j, k+1-j\})$, де $j \neq i$, визначити реброву функцію, яка представляє собою покриває функцію для поточного шаблону:

$$f_j = D_{n_1}^j(\{x_1, \dots, x_{n_1}\}) \cap D_{n_2}^{k+1-j}(\{x_{n_1+1}, \dots, x_{n_1+n_2}\}).$$

Числові параметри моделі:

e – кількість елементів в моделі;

d – сумарна кількість діз'юнктивних операцій;

c – сумарна кількість кон'юнктивні операцій;

a – кількість функцій;

$$k = K + 1,$$

де k – кількість відмовили елементів, при якому система стає непрацездатною;

де K – кількість відмовили елементів, при якому система ще залишається працездатною;

$$1) \quad e = \sum_{l=0}^k (l * C_p^l + (k-l) * C_q^{k-l}),$$

$$d = \sum_{l=0}^k (C_p^l + C_q^{k-l} - 2),$$

$$c = \sum_{l=0}^k ((l-1) * C_p^l + (k-l-1) * C_q^{k-l} + 1) - 1,$$

$$a = k + 2,$$

при $p \geq k, q \geq k$

Як і для 2-відмовостійкої моделі, $e = d + c + a$.

$$2) \quad e = \sum_{l=k-q}^p (l * C_p^l + (k-l) * C_q^{k-l}),$$

$$d = \sum_{l=k-q}^p (C_p^l + C_q^{k-l} - 2),$$

$$c = \sum_{l=k-q}^p ((l-1) * C_p^l + (k-l-1) * C_q^{k-l} + 1) - 1,$$

$$a = n - k + 2,$$

при $p \leq k, q \leq k$

Аналогічно, $e = d + c + a$.

3.2.2 Приклад.

Розглянемо ВБС, захищену від відмов кратності не вище 5, що складається з 9 елементів (модулів). $D_9^6(\{x_1, \dots, x_9\})$ - безліч всіх критичних комбінацій такої ВБС.

Запишемо шаблони для даного варіанту:

$$\kappa_1 = K(1,4) \times K(5,5),$$

$$\kappa_2 = K(2,4) \times K(4,5),$$

$$\kappa_3 = K(3,4) \times K(3,5),$$

$$\kappa_4 = K(4,4) \times K(2,5).$$

$$k_1 = D_4^1(\{x_1 \dots x_4\}) * D_5^5(\{x_6 \dots x_9\})$$

$$k_2 = D_4^2(\{x_1 \dots x_4\}) * D_5^4(\{x_6 \dots x_9\})$$

$$k_3 = D_4^3(\{x_1 \dots x_4\}) * D_5^3(\{x_6 \dots x_9\})$$

$$k_4 = D_4^4(\{x_1 \dots x_4\}) * D_5^2(\{x_6 \dots x_9\})$$

Розбиваємо k_l на дві функції і отримуємо булеві реберні функції.

$$f_1 = D_4^3(\{x_1 \dots x_4\})$$

$$f_2 = D_5^3(\{x_6 \dots x_9\})$$

$$f_3 = D_4^2(\{x_1 \dots x_4\}) * D_5^4(\{x_6 \dots x_9\})$$

$$f_4 = D_4^1(\{x_1 \dots x_4\}) * D_5^5(\{x_6 \dots x_9\})$$

$$f_5 = D_4^4(\{x_1 \dots x_4\}) * D_5^2(\{x_6 \dots x_9\})$$

Перевіряємо, чи правильно сформували функції. Критерієм правильності формування є рівність множини $D_9^6(\{x_1, \dots, x_9\})$ і множина, що отримана перемноживши всі реберні функції між собою.

$$f_1 * (f_2 + f_3 + f_4 + f_5) + f_2 * (f_3 + f_4 + f_5) + f_3 * (f_4 + f_5) + (f_4 * f_5) =$$

$$\begin{aligned}
& D_4^3(\{x_1..x_4\}) * (D_5^3(\{x_6..x_9\}) + D_4^2(\{x_1..x_4\}) * D_5^4(\{x_6..x_9\}) + D_4^1(\{x_1..x_4\}) * \\
& D_5^5(\{x_6..x_9\}) + D_4^4(\{x_1..x_4\}) * D_5^2(\{x_6..x_9\}) + D_5^3(\{x_6..x_9\}) * (D_4^2(\{x_1..x_4\}) \\
& * D_5^4(\{x_6..x_9\}) + D_4^1(\{x_1..x_4\}) * D_5^5(\{x_6..x_9\}) + D_4^4(\{x_1..x_4\}) * D_5^2(\{x_6..x_9\}) \\
&) + D_4^2(\{x_1..x_4\}) * D_5^4(\{x_6..x_9\}) * (D_4^1(\{x_1..x_4\}) * D_5^5(\{x_6..x_9\}) + \\
& D_4^4(\{x_1..x_4\}) * D_5^2(\{x_6..x_9\})) + D_4^1(\{x_1..x_4\}) * D_5^5(\{x_6..x_9\}) * D_4^4(\{x_1..x_4\}) * \\
& D_5^2(\{x_6..x_9\}) = D_4^1(\{x_1..x_4\}) * D_5^5(\{x_6..x_9\}) + D_4^2(\{x_1..x_4\}) * D_5^4(\{x_6..x_9\}) + \\
& D_4^3(\{x_1..x_4\}) * D_5^3(\{x_6..x_9\}) + D_4^4(\{x_1..x_4\}) * D_5^2(\{x_6..x_9\}) = \\
& D_9^6(\{x_1,..,x_9\})
\end{aligned}$$

Сумарна кількість елементів функцій в не мінімізованої моделі обчислюється за формулою 2 (при $p \leq k, q \leq k$) :

$$\begin{aligned}
e &= \sum_{l=k-q}^p (l * C_p^l + (k-l) * C_q^{k-l}), \\
d &= \sum_{l=k-q}^p (C_p^l + C_q^{k-l} - 2), \\
c &= \sum_{l=k-q}^p ((l-1) * C_p^l + (k-l-1) * C_q^{k-l} + 1) - 1, \\
a &= n - k + 2
\end{aligned}$$

У нашому випадку: $p=5, q=4, k=6$

$$\begin{aligned}
e &= \sum_{l=2}^5 (l * C_5^l + (6-l) * C_4^{6-l}) = 107, \\
d &= \sum_{l=2}^5 (C_5^l + C_4^{6-l} - 2) = 33,
\end{aligned}$$

$$c = \sum_{l=2}^5 \left((l-1) * C_5^l + (5-l) * C_4^{6-l} + 1 \right) - 1 = 69,$$

$$a = 5.$$

$$e = d + c + a.$$

3.2.3 Мінімізація GL-моделі системи, стійкої до відмов довільної кратності

e' – кількість елементів в не мінімізованій моделі;

d' – сумарна кількість діз'юнктивних операцій;

c' – сумарна кількість кон'юнктивних операцій;

$$k = K + 1,$$

де k – кількість відмовили елементів, при якому система стає непрацездатною;

де K – кількість відмовили елементів, при якому система ще залишається працездатною;

$$1) \quad p \geq k, q \geq k$$

$$2) \quad p \leq k, q \leq k$$

Перетворення $D_n^k(\{x_1 \dots x_n\})$ до вигляду 1) або 2), в залежності від співвідношення k с m і $(n-m)$:

$$\bigcup_{l=0}^k D_m^l(\{x_1 \dots x_m\}) \cap D_{n-m}^{k-l}(\{x_{m+1} \dots x_n\}), \text{ при } m \geq k, n-m \geq k \quad 1)$$

$$\bigcup_{l=k-m}^{n-m} D_m^{k-l}(\{x_1 \dots x_m\}) \cap D_{n-m}^l(\{x_{m+1} \dots x_n\}), \text{ при } m \leq k, n-m \leq k$$

2)

На даному етапі маємо зменшення складності еквівалентної $D_n^k(\{x_1 \dots x_n\})$ моделі.

Поняття еквівалентності має на увазі ідентичність набору критичних комбінацій.

За заданою диз'юнкції формуються реберні функції.

Перші дві функції формуються для $l=l_s = \lfloor k/2 \rfloor$ для 1)

або $l=l_s = \lfloor (n-k)/2 \rfloor$ для випадку 2).

Набір функцій для першого випадку 1):

$$F_1 = D_m^{l_s}(\{x_1 \dots x_m\})$$

$$F_2 = D_{n-m}^{k-l_s}(\{x_{m+1} \dots x_n\})$$

$$F_3 = D_m^{l_s+1}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s-1}(\{x_{m+1} \dots x_n\})$$

$$F_4 = D_m^{l_s+2}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s-2}(\{x_{m+1} \dots x_n\})$$

...

$$F_{k-l_s+2} = D_m^k(\{x_1 \dots x_m\}) * D_{n-m}^0(\{x_{m+1} \dots x_n\})$$

$$F_{k-l_s+3} = D_m^{l_s-1}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s+1}(\{x_{m+1} \dots x_n\})$$

$$F_{k-l_s+4} = D_m^{l_s-2}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s+2}(\{x_{m+1} \dots x_n\})$$

...

$$F_{k+2} = D_m^0(\{x_1 \dots x_m\}) * D_{n-m}^k(\{x_{m+1} \dots x_n\})$$

Набір функцій для другого випадку – 2):

$$F_1 = D_m^{l_s}(\{x_1 \dots x_m\})$$

$$F_2 = D_{n-m}^{k-l_s}(\{x_{m+1} \dots x_n\})$$

$$F_3 = D_m^{l_s+1}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s-1}(\{x_{m+1} \dots x_n\})$$

$$F_4 = D_m^{l_s+2}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s-2}(\{x_{m+1} \dots x_n\})$$

...

$$F_{m-l_s+2} = D_m^m(\{x_1 \dots x_m\}) * D_{n-m}^{k-m}(\{x_{m+1} \dots x_n\})$$

$$F_{m-l_s+3} = D_m^{l_s-1}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s+1}(\{x_{m+1} \dots x_n\})$$

$$F_{m-l_s+4} = D_m^{l_s-2}(\{x_1 \dots x_m\}) * D_{n-m}^{k-l_s+2}(\{x_{m+1} \dots x_n\})$$

...

$$F_{n-k+2} = D_m^{k+m-n}(\{x_1 \dots x_m\}) * D_{n-m}^{n-m}(\{x_{m+1} \dots x_n\})$$

Дані функції розташовуються на ребрах кільцевого графа.

Тепер стоїть завдання мінімізації реберних функцій.

Мінімізація $D_n^{k'}(\{x_1 \dots x_n\})$ складається з двох етапів.

Перший етап, ідентичний розбиття $D_n^k(\{x_1 \dots x_n\})$ на функції, в залежності від співвідношення k' с m' і $(n'-m')$:

$$\bigcup_{l=0}^{k'} D_{m'}^l(\{x_1 \dots x_{m'}\}) \cap D_{n'-m'}^{k'-l}(\{x_{m'+1} \dots x_{n'}\}), \text{ при } m' \geq k', n'-m' \geq k' \quad 3)$$

$$\bigcup_{l=k'-m'}^{n'-m'} D_{m'}^{k'-l}(\{x_1 \dots x_{m'}\}) \cap D_{n'-m'}^l(\{x_{m'+1} \dots x_{n'}\}), \text{ при } m' \leq k', n'-m' \leq k' \quad 4)$$

Аналогічно, перші дві функції формуються для $l=l_s' = \lfloor k'/2 \rfloor$ для 3)

або $l=l_s' = \lfloor (n'-k')/2 \rfloor$ для випадку 4).

Набір функцій для першого випадку 3):

$$F_1' = D_{m'}^{l_s'}(\{x_1 \dots x_{m'}\})$$

$$F_2' = D_{n'-m'}^{k'-l_s'}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_3' = D_{m'}^{l_s'+1}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'-1}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_4' = D_{m'}^{l_s'+2}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'-2}(\{x_{m'+1} \dots x_{n'}\})$$

...

$$F_{k'-l_s'+2}' = D_{m'}^{k'}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^0(\{x_{m'+1} \dots x_{n'}\})$$

$$F_{k'-l_s'+3}' = D_{m'}^{l_s'-1}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'+1}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_{k'-l_s'+4}' = D_{m'}^{l_s'-2}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'+2}(\{x_{m'+1} \dots x_{n'}\})$$

...

$$F_{k+2}' = D_{m'}^0(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'}(\{x_{m'+1} \dots x_{n'}\})$$

Набір функцій для другого випадку – 4):

$$F_1' = D_{m'}^{l_s'}(\{x_1 \dots x_{m'}\})$$

$$F_2' = D_{n'-m'}^{k'-l_s'}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_3' = D_{m'}^{l_s'+1}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'-1}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_4' = D_{m'}^{l_s'+2}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'-2}(\{x_{m'+1} \dots x_{n'}\})$$

...

$$F_{m'-l_s'+2}' = D_{m'}^{m'}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-m'}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_{m'-l_s'+3}' = D_{m'}^{l_s'-1}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'+1}(\{x_{m'+1} \dots x_{n'}\})$$

$$F_{m'-l_s'+4}' = D_{m'}^{l_s'-2}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l_s'+2}(\{x_{m'+1} \dots x_{n'}\})$$

...

$$F'_{m'-k'+2} = D_{m'}^{k'+m'-n'}(\{x_1..x_{m'}\}) * D_{n'-m'}^{n'-m'}(\{x_{m'+1}..x_{n'}\})$$

Зазначене подання $D_n^{k'}(\{x_1..x_{n'}\})$, у вигляді сукупності F' функцій, необхідно представити у вигляді кон'юнкції диз'юнкцій F' функцій, тобто у вигляді однієї функції G.

Другий етап мінімізації:

$$G = K_1 * K_2$$

Знову ж, є два варіанти подання функції G, для $m' \geq k'$, $n'-m' \geq k'$ і для $m' \leq k'$, $n'-m' \leq k'$.

$$1) m' \geq k', n'-m' \geq k'$$

(*)

$$K_1 = \bigcup_{i=1}^{k'-l'_s} D_{m'}^{l'_s+i}(\{x_1..x_{m'}\}) * D_{n'-m'}^{k'-l'_s-i}(\{x_{m'+1}..x_{n'}\}) \cup D_{n'-m'}^{k'-l'_s}(\{x_{m'+1}..x_{n'}\})$$

$$K_2 = \bigcup_{i=1}^{l'_s} D_{m'}^{l'_s-i}(\{x_1..x_{m'}\}) * D_{n'-m'}^{k'-l'_s+i}(\{x_{m'+1}..x_{n'}\}) \cup D_{m'}^{l'_s}(\{x_1..x_{m'}\})$$

Розіб'ємо твір K_1 на K_2 на добуток їх складових.

$$D_{n'-m'}^{k'-l'_s}(\{x_{m'+1}..x_{n'}\}) * K_2 =$$

$$\begin{aligned}
& D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) * \left(\bigcup_{i=1}^{l'_s} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s+i}(\{x_{m'+1} \dots x_{n'-m'}\}) \cup D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) \right) = \\
& \bigcup_{i=1}^{l'_s} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s+i}(\{x_{m'+1} \dots x_{n'-m'}\}) \cup D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'-m'}\}) \\
& D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) * K_1 = \\
& D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) * \left(\bigcup_{i=1}^{k'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s-i}(\{x_{m'+1} \dots x_{n'-m'}\}) \cup D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'-m'}\}) \right) = \\
& \bigcup_{i=1}^{k'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s-i}(\{x_{m'+1} \dots x_{n'-m'}\}) \cup D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'-m'}\})
\end{aligned}$$

Добуток

$$\left(\bigcup_{i=1}^{l'_s} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \right) * \left(\bigcup_{i=1}^{k'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \right)$$

,

так як

$$D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \in D_{n'-m'}^{k'-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \quad \text{для} \quad \text{будь-якого} \quad i \in (1; l'_s) \quad -$$

поглинання.

Отримуємо:

$$\begin{aligned}
G = K_1 * K_2 &= \left(\bigcup_{i=1}^{k'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \right) * \\
&* \left(\bigcup_{i=1}^{l'_s} D_m^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_m^{l'_s}(\{x_1 \dots x_{m'}\}) \right) = \\
&= \bigcup_{i=1}^{l'_s} D_m^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup \bigcup_{i=1}^{k'-l'_s} D_m^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup \\
&\cup D_m^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) = D_n^{k'}(\{x_1 \dots x_{n'}\})
\end{aligned}$$

$$\begin{aligned}
&2) m' \leq k', n'-m' \leq k' \\
&(**)
\end{aligned}$$

$$\begin{aligned}
K_1 &= \bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \\
K_2 &= \bigcup_{i=1}^{l'_s-k'} D_m^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_m^{l'_s}(\{x_1 \dots x_{m'}\})
\end{aligned}$$

Розіб'ємо добуток K_1 на K_2 на добуток їх складових.

$$\begin{aligned}
&D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) * K_2 = \\
&D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) * \left(\bigcup_{i=1}^{l'_s-k'} D_m^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_m^{l'_s}(\{x_1 \dots x_{m'}\}) \right) = \\
&\bigcup_{i=1}^{l'_s-k'} D_m^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_m^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \\
&D_m^{l'_s}(\{x_1 \dots x_{m'}\}) * K_1 = \\
&D_m^{l'_s}(\{x_1 \dots x_{m'}\}) * \left(\bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \right) = \\
&\bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_m^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k'-l'_s}(\{x_{m'+1} \dots x_{n'}\})
\end{aligned}$$

Добуток

$$\left(\bigcup_{i=1}^{l'_s-k'} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \right) * \left(\bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \right)$$

, так як

$D_{n'-m'}^{k-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \in D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\})$ для будь-якого $i \in (1; l'_s - k')$ - поглинання.

Аналогічно:

$$\begin{aligned} G = K_1 * K_2 &= \left(\bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_{n'-m'}^{k-l'_s}(\{x_{m'+1} \dots x_{n'}\}) \right) \\ &* \left(\bigcup_{i=1}^{l'_s-k'} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) \right) = \\ &= \bigcup_{i=1}^{l'_s-k'} D_{m'}^{l'_s-i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s+i}(\{x_{m'+1} \dots x_{n'}\}) \cup \bigcup_{i=1}^{m'-l'_s} D_{m'}^{l'_s+i}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s-i}(\{x_{m'+1} \dots x_{n'}\}) \cup \\ &\cup D_{m'}^{l'_s}(\{x_1 \dots x_{m'}\}) * D_{n'-m'}^{k-l'_s}(\{x_{m'+1} \dots x_{n'}\}) = D_n^{k'}(\{x_1 \dots x_{n'}\}) \end{aligned}$$

Той факт, що для двох випадків $G = D_n^{k'}(\{x_1 \dots x_{n'}\})$, для різного співвідношення k' с m' і $n'-m'$, говорить про те, що $D_n^{k'}(\{x_1 \dots x_{n'}\})$ в реберної функції можна замінити на добуток $K_1 * K_2$, де K_1 і K_2 мають строго певний вид, зазначений в (*) і (**).

Пункти мінімізації 1) і 2) необхідно повторювати до тих пір, поки k' не дорівнюватиме 1 для розбиваємого $D_n^{k'}(\{x_1 \dots x_{n'}\})$, тобто до диз'юнкцій виду $D_n^1(\{x_1 \dots x_{n'}\})$.

3.2.4 Приклад мінімізації.

$$f_1 = D_4^3(\{x_1..x_4\})$$

$$f_2 = D_5^3(\{x_5..x_9\})$$

$$f_3 = D_4^2(\{x_1..x_4\}) * D_5^4(\{x_5..x_9\})$$

$$f_4 = D_4^1(\{x_1..x_4\}) * D_5^5(\{x_5..x_9\})$$

$$f_5 = D_4^4(\{x_1..x_4\}) * D_5^2(\{x_5..x_9\})$$

Почнемо мінімізацію $D_4^3(\{x_1..x_4\})$:

$$f_{11} = D_2^1(\{x_1..x_2\})$$

$$f_{21} = D_2^2(\{x_3..x_4\})$$

$$f_{31} = D_2^2(\{x_1..x_2\}) * D_2^1(\{x_3..x_4\})$$

Позначимо через $|f|$ сумарну кількість елементів системи, які описують функцію f .

$|f_{11}| + |f_{21}| + |f_{31}| = 2 + 2 + 4 = 8$, а $|D_4^3(\{x_1..x_4\})| = 12$ – мінімізація очевидна.

$$D_4^3(\{x_1..x_4\}) = f_{11} * (f_{21} + f_{31}) = (x_1 + x_2) * (x_3x_4 + x_1x_2 * (x_3 + x_4))$$

Мінімізація $D_5^3(\{x_5..x_9\})$:

$$f_{12} = D_2^1(\{x_5..x_6\})$$

$$f_{22} = D_3^2(\{x_7..x_9\})$$

$$f_{32} = D_2^2(\{x_5..x_6\}) * D_3^1(\{x_7..x_9\})$$

$$f_{42} = D_3^3(\{x_7..x_9\})$$

Тепер треба мінімізувати $f_{22} = D_3^2(\{x_7..x_9\})$:

$$f'_{12} = x_7$$

$$f'_{22} = D_2^1(\{x_8..x_9\})$$

$$f'_{32} = D_2^2(\{x_8..x_9\})$$

Мінімізовану f_{22} :

$$f_{22} = D_3^2(\{x_7..x_9\}) = (f'_{12} + f'_{32}) * f'_{22} = (x_7 + x_8x_9) * (x_8 + x_9)$$

$$D_5^3(\{x_5..x_9\}) = (f_{12} + f_{42}) * (f_{22} + f_{32}) =$$

$$(x_5 + x_6 + x_7x_8x_9) * ((x_7 + x_8x_9) * (x_8 + x_9) + x_5x_6 * (x_7 + x_8 + x_9))$$

$$|(x_5 + x_6 + x_7x_8x_9) * ((x_7 + x_8x_9) * (x_8 + x_9) + x_5x_6 * (x_7 + x_8 + x_9))|$$

=15

Не мінімізована $|D_5^3(\{x_5..x_9\})| = 30$.

Мінімізація $D_4^2(\{x_1..x_4\})$:

$$f_{13} = D_2^1(\{x_1..x_2\})$$

$$f_{23} = D_2^1(\{x_3..x_4\})$$

$$f_{33} = D_2^2(\{x_1..x_2\})$$

$$f_{43} = D_2^2(\{x_3..x_4\})$$

$$D_4^2(\{x_1..x_4\})=(f_{13}+f_{43})*(f_{23}+f_{33})=$$

$$(x_1+x_2+x_3x_4)*(x_3+x_4+x_1x_2)$$

$$\text{Мінімізована } |(x_1+x_2+x_3x_4)*(x_3+x_4+x_1x_2)|=8.$$

$$\text{Не мінімізована } |D_4^2(\{x_1..x_4\})|=12.$$

$$\text{Мінімізація } D_5^4(\{x_5..x_9\}):$$

$$f_{14}=D_2^2(\{x_5..x_6\})$$

$$f_{24}=D_3^2(\{x_7..x_9\})$$

$$f_{34}=D_2^1(\{x_5..x_6\})*D_3^3(\{x_7..x_9\})$$

$$D_3^2(\{x_7..x_9\})=x_7*(x_8+x_9)+x_8x_9$$

$$D_5^4(\{x_6..x_9\})=(f_{14}+f_{34})*f_{24}=$$

$$(x_5x_6+(x_5+x_6)*x_7x_8x_9)*(x_7*(x_8+x_9)+x_8x_9)$$

$$\text{Мінімізована}$$

$$|(x_5x_6+(x_5+x_6)*x_7x_8x_9)*(x_7*(x_8+x_9)+x_8x_9)|=12.$$

$$\text{Не мінімізована } |D_5^4(\{x_6..x_9\})|=20.$$

$$\text{Мінімізація } D_5^2(\{x_5..x_9\}):$$

$$f_{15}=D_2^1(\{x_5..x_6\})$$

$$f_{25}=D_3^1(\{x_7..x_9\})$$

$$f_{35}=D_2^2(\{x_5..x_6\})$$

$$f_{45} = D_3^2(\{x_7 \dots x_9\})$$

$$D_3^2(\{x_7 \dots x_9\}) = x_7 * (x_8 + x_9) + x_8 x_9$$

$$D_5^2(\{x_5 \dots x_9\}) = (f_{15} + f_{45}) * (f_{25} + f_{35}) = \\ (x_5 + x_6 + x_7 * (x_8 + x_9) + x_8 x_9) * (x_7 + x_8 + x_9 + x_5 x_6)$$

Мінімізована

-

$$|(x_5 + x_6 + x_7 * (x_8 + x_9) + x_8 x_9) * (x_7 + x_8 + x_9 + x_5 x_6)| = 12.$$

$$\text{Не мінімізована} - |D_5^2(\{x_5 \dots x_9\})| = 20.$$

Запис вже мінімізованих $f_1 \dots f_5$:

$$f_1 = (x_1 + x_2) * (x_3 x_4 + x_1 x_2 * (x_3 + x_4))$$

$$f_2 =$$

$$(x_5 + x_6 + x_7 x_8 x_9) * ((x_7 + x_8 x_9) * (x_8 + x_9) + x_5 x_6 * (x_7 + x_8 + x_9))$$

$$f_3 = ((x_1 + x_2 + x_3 x_4) * (x_3 + x_4 + x_1 x_2)) *$$

$$((x_5 x_6 + (x_5 + x_6) * x_7 x_8 x_9) * (x_7 * (x_8 + x_9) + x_8 x_9))$$

$$f_4 = (x_1 + x_2 + x_3 + x_4) * x_5 x_6 x_7 x_8 x_9$$

$$f_5 = x_1 x_2 x_3 x_4 *$$

$$((x_5 + x_6 + x_7 * (x_8 + x_9) + x_8 x_9) * (x_7 + x_8 + x_9 + x_5 x_6))$$

Характеристики мінімізованої системи функцій $f_1 \dots f_5$:

$$e' = |f_1| + |f_2| + |f_3| + |f_4| + |f_5| = 8 + 15 + (8 + 12) + (4 + 5) + (4 + 12) = 68.$$

$$d' = 28$$

$$c' = 35$$

Характеристики не мінімізованої моделі:

$e=107$

$d=33$

$c=69$

3.2.5 Висновки

Розглянуто базові підсистеми ВБС і два методи побудови моделей їх поведінки в потоці відмов, доведені необхідні теореми.

Перший метод порівняно нескладний, передбачає просту процедуру формування вираження функції f_{i+1} , виходячи з виразу для функції f_i , число ребер завжди дорівнює числу модулів системи і дає можливість формування моделей для 2-стійких ВБС.

Другий метод дозволяє будувати K_n^m -моделі ВБС для довільних значень m і n . Число ребер GL-моделі, як правило, перевищує число процесорів системи, але рівняння для ребрових функцій порівняно нескладні і неповторні. Розроблено спрощену процедуру формування GL-моделі ВБС, заснована на використанні простих виразів функцій для випадків K_i^1 і K_i^i . Процедура передбачає розбиття множини модулів ВБС на підмножини і представлення моделі за допомогою таблиць, осередками яких є моделі меншої розмірності.

Запропоновано методику синтезу базових циклічних моделей для випадку, коли реберні функції представляються у вигляді кон'юнктивні нормальних форм. Розглянуто особливості таких моделей і їх мінімізація.

4. ШЛЯХИ ПЕРЕТВОРЕННЯ ЦИКЛІЧНИХ GL-МОДЕЛЕЙ

4.1 Внутрішні ребра в графі GL-моделі

Розроблені варіанти GL-моделей, що названі базовими, дозволяють моделювати поведінку системи, яка зберігає працездатність при обмеженій кількості відмов, не перевищуючій фіксовану величину $m < n$ (де n – число модулів в системі). Такі системи і їх моделі будемо позначати K_n^m .

Проте в реальних ВБС часто стається так, що система продовжує функціонувати і при виході зі строю деяких множин відмовивших модулів, де число останніх перевищує величину m . Наприклад, ВБС може реконфігуруватись в більшості випадків 2-кратних відмов її модулів, проте на визначеній сукупності векторів стану модулів системи, що мають два нулі, вона відмовляє, залишаючись в той же час працездатною на деяких сукупностях векторів станів, що мають 3 або навіть 4 одиниці. Такі вимоги диктуються конкретними поглядами, архітектурою, конфігурацією системи, характеристиками модулів в процесі реконфігурації та іншими факторами. Звісно, для аналізу поведінки таких систем потрібні складніші модулі. Цю задачу ми і будемо вирішувати в даній роботі, яка складається з трьох розділів. В перших двох ми будемо розглядати, залишаючись в рамках GL-моделей, дві можливості трансформації базових моделей: перетворення булевих функцій, що приписуються ребрам циклічного графу, і введення додаткових ребер. В третьому розділі ми займемося рішенням задачі оптимізації проведення цих додаткових ребер.

Розглянемо обидві вище вказані можливості. При цьому зробимо наступне, майже очевидне зауваження: для будь-якої заданої сукупності векторів стану системи, для яких потрібно вирішувати поставлену задачу, можлива покрокова зміна GL-моделі (вектор за вектором), якщо не

порушується її адекватність для всієї, вже розглянутої до цього кроку, сукупності наборів. Керуючись цим зауваженням, спростимо задачу.

Нехай є базова система K_n^m і її GL-модель. Необхідно побудувати GL-модель для системи, яка, маючи властивості базової K_n^m (тобто вона стійка до відмов не більше m своїх модулів), залишається працездатною ще і на векторі станів α :

$$\begin{array}{cccccccccccc} x_1 & x_2 & x_3 & \dots & x_m & x_{m+1} & x_{m+2} & \dots & x_n \\ 1 & 1 & 1 & \dots & 1 & 1 & 0 & \dots & 0 \end{array}$$

Тут, x_i - модуль системи, який може бути в стані – справний, або 1 – несправний. Іншими словами, наша модернізована GL-модель не повинна втрачати зв'язності при появі вектору α .

4.1.1 Реберні функції.

Побудуємо базову GL-модель K_n^m . У відповідності з методикою, викладеною в [1], відмітимо, що при появі вектору станів α не менш двох функцій, що приписуються ребрам циклічного графу, приймуть нульове значення. В графі GL-моделі зникнуть два ребра, і він втратить зв'язність, тобто GL-модель K_n^m не адекватна системі, яку ми маємо побудувати. Нехай для конкретності це будуть рівно дві функції: f_1 і f_2 .

Справедливо наступне: для збереження адекватності GL-моделі на векторі α достатньо ребру з функцією f_1 приписати функцію $f_1^* = f_1 \vee H$, де H – конститuenta нуля, відповідна до набору α :

$$H = \bar{x}_1 + \bar{x}_2 + \dots + \bar{x}_{m+1} + x_{m+2} + \dots + x_n$$

Дійсно, ребро графу з новою функцією f_1^* при появі вектору α не зникає ($f_1^* \neq 0$), а отже, зв'язність графу зберігається. При появі інших векторів стану функція f_1^* поводить себе так же, як функція f_1 , і, тому, адекватність GL-моделі не порушується. Якщо ж при появі вектора α на вихідній базовій GL-моделі зникає більше, ніж два ребра одночасно, достатньо всі функції цих ребер, окрім одної, змінити таким же д образом,

тобто додати ту ж конститuentу одиниці H . Зрозуміло, що в цьому випадку саме на наборі α зв'язність графу не порушиться (випадає лише одне ребро).

Тут варто відмітити, що одне і те ж ребро може зникати при появі багатьох, наприклад s , векторів станів на множину, яка задана. Якщо s відповідних конститuent одиниці додати до функції f_l , приписаної цьому ребру, то отриманий вираз

$$f_l^* = f_l \vee H_1 \vee H_2 \vee \dots \vee H_s$$

може бути мінімізований любим з відомих методів мінімізації форм представлення булевих функцій, що ніяк не відобразиться на адекватності GL-моделі. Зрозуміло, що вибір конститuent одиниці, що приписуються тим чи іншим ребрам, може суттєво вплинути на кінцеву форму представлення реберних функцій.

Відмітимо тут, що розглянутий підхід є універсальним в тому сенсі, що дозволяє будувати GL-моделі як для систем з пониженим ступенем відмовостійкості, так і для систем з підвищеним ступенем відмовостійкості по відношенню до базових GL-моделей.

4.1.2 Додаткові ребра.

Покажемо, що вирішення задачі, поставленої в 2, шляхом введення в базову GL-модель додаткових ребер можливо. Зробимо це конструктивним шляхом.

Перш за все відмітимо наступне.

Додаткове ребро r з функцією f , що з'єднує дві які-небудь вершини GL – моделі, змінює останню тільки в одному напрямку: нова GL-модель адекватна системі з вищим ступенем відмовостійкості.

Насправді, функція f може приймати два значення: 0 і 1. Одиничне значення функції, тобто відсутність додаткового ребра, не міняє зв'язності

вихідного графу. Нульове значення функції f підвищує «ступінь» зв'язності, так як з'являються нові шляхи на графі.

Далі:

1. Якщо при появі заданого вектору станів α випадають ребра a_i і a_j , то додаткове ребро r має проходити між ними.

Очевидно, що в іншому випадку порушення зв'язності графу GL-моделі відбудеться незалежно від функції, що приписується ребру r .

2. Якщо при появі заданого вектору станів α випадає декілька ребер $a_1, a_2 \dots a_k$, то, ребра $r_1, r_2 \dots r_L$, що вводяться в базову GL-модель, мають бути приведені таким чином, щоб між двома найближчими вершинами, інцидентними двом довільним ребрам, знаходилось не більше одного ребра $a_i \in \{a_1, a_2 \dots a_k\}$ вихідного графу.

Насправді, в іншому випадку два ребра $a_i, a_j \in \{a_1, a_2 \dots a_k\}$, що знаходяться між вказаними вершинами, утворюють ізольовану частину графу при появі вектору α , і, отже, зв'язність графу порушиться.

3. Для збереження зв'язності графу при появі вектору α достатньо ребру r (або ребрам $r_1 \dots r_L$), побудованим вказаним образом, приписати функцію H , тобто ту ж конституюнту нуля.

Дійсно, таке ребро r (або ребра $r_1 \dots r_L$ з однією і тою ж функцією) запобігають втраті зв'язності в графі, оскільки забезпечують шлях на графі між двома ділянками кільця, що роз'єднані зникнувшими ребрами. З іншої сторони, воно не змінює GL-модель на інших векторах, так як воно відсутнє при появі всіх векторів станів, крім α ($H = 0$ тільки на одному наборі).

Нехай в базовій GL-моделі для вектору α проведено додаткове ребро r_1 з функцією H_1 . Нехай існує другий вектор стану β , для якого необхідно вирішити таку ж задачу. Тоді, якщо ребра a_i і a_j , що випадають при появі вектора β , лежать по різні сторони від ребра r_1 , достатньо, для збереження

GL-моделі приписати ребру r_1 функцію $f = H_1 \vee H_2$, де H_2 - конститuenta нуля, що відповідає вектору β . Якщо ребра a_i і a_j лежать по одну сторону ребра r_1 , то для вирішення задачі необхідно інше додаткове r_2 , яке слід проводити, керуючись тими ж правилами.

Іншими словами, кожному додатковому ребру приводиться у відповідність функція, яка є диз'юнкцією конститuent тих векторів стану модулів системи, відносно яких саме це додаткове ребро на GL-моделі має запобігати втраті зв'язності.

Реберна функція фактично являється частично визначеною. В частості, її значення на векторах станів, поява яких призводить до зникненню ребер базової GL-моделі, що знаходяться з одної сторони ребра r_1 , може бути встановлено довільно, оскільки не впливає на функціонування GL-моделі.

Для ілюстрації сказаного розглянемо наступний приклад.

Побудуємо GL-модель 4-модульної системи, яка стійка до всіх одиночних відмов і, окрім того до відмов груп модулів $\{1,3\}$ і $\{2,3\}$.

Будуємо K_4^1 GL-модель (див. рис 4.1a). Для даного прикладу достатньо одного внутрішнього ребра (рис. 4.1b). Функція f визначається легко:

$$f = (\bar{x}_1 \vee \bar{x}_3 \vee x_2 \vee x_4) * (\bar{x}_2 \vee \bar{x}_3 \vee x_1 \vee x_4)$$

Враховуючи також, що, як зазначалось вище, функція f може приймати довільні значення на всіх наборах виду $11xx$ і $xx11$ (тут x означає будь-яке значення 0 або 1). Використовуючи відомі методи мінімізації не повністю визначених булевих функцій, отримуємо:

$$f = \bar{x}_3 \vee \bar{x}_1 \bar{x}_2$$

В даному випадку можна відмітити, що функція f не визначена також на наборах 0001, 0010, 0100, 1000, тобто на векторах станів системи, коли із строю виходить не більше одного модуля. Це дозволяє спростити її ще більше: $f = \bar{x}_3$

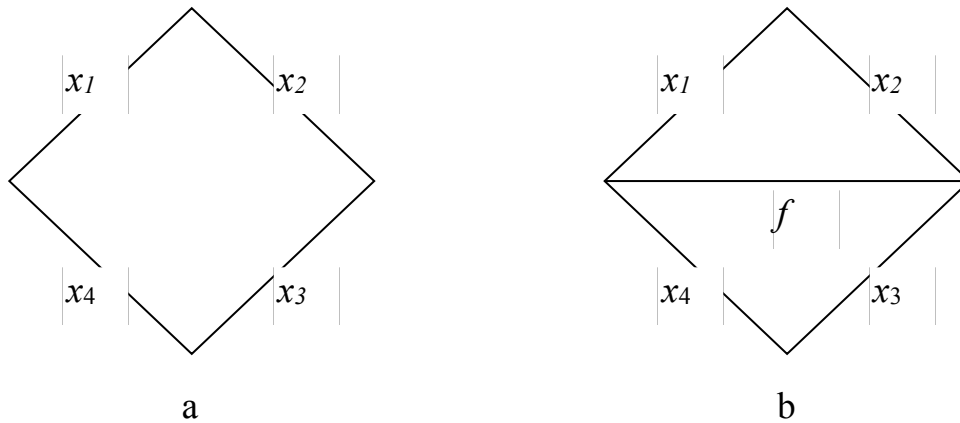


Рис. 4.1

Відмітимо тут, що кожне нове додаткове ребро, побудоване згідно правилам, описаним вище, змінює GL-модель лише для одного вектору α станів модулів ВБС, залишаючи всі властивості і характеристики незмінними для всіх інших векторів. Додаткових ребер може бути декілька. Їх кількість визначається потужністю множини векторів станів ВБС, яка перевищує множину базової системи, де ВБС зберігає працездатність.

4.2 Заміна реберних функцій GL-моделі

Так же, як і в попередньому випадку, можемо відмітити два шляхи вирішення поставленої в заголовку задачі: зміна функцій, що приписуються ребрам, і введення додаткових ребер. Аналогічно розглянутому в 2 припускаємо, що є базова система K_n^m і її GL-модель, але, крім того, при появі вектору станів α :

$$\begin{array}{cccccccc} x_1 & x_2 & \dots & x_{m-1} & x_m & x_{m+1} & \dots & x_n \\ 1 & 1 & \dots & 1 & 1 & 0 & \dots & 0 \end{array}$$

система відмовляє, тобто в моделі має місце втрата зв'язності.

4.2.1 Зміна реберних функцій.

Покажемо наступне:

Для того, що трансформувати базову GL-модель таким образом, що вона адекватно буде відображати пониження степеня відмовостійкості системи на векторі станів α , достатньо функцію f_j якого-небудь ребра a_j замінити на функцію:

$$f_j^* = f_j \vee x_1 * x_2 * \dots * x_m$$

Дійсно, нехай поява вектору α викликає зникнення одного ребра a_i . Для того, щоб GL-модель відображала відмову системи, необхідно, щоб зникло ще хоча б одне ребро. Цього можна досягнути, якщо додати до функції f_j любого ребра a_j , $j \neq i$ елементарну диз'юнкцію

$$\varphi = x_1 * x_2 * \dots * x_m * \bar{x}_{m+1} * \dots * \bar{x}_n,$$

тобто на конститuentу одиниці. Проте в розглянутому випадку (вектор α містить m нулів) функція φ може бути спрощена.

$$\varphi = x_1 * x_2 * \dots * x_m$$

Це можливо по наступним причинам:

1. Якщо в реальному векторі стану всі компоненти системи крім $x_1 \dots x_m$, справні, то вираз $\bar{x}_{m+1} * \dots * \bar{x}_n$ може бути відкинуто по властивостям диз'юнкції;
2. Якщо в реальному векторі стану хоч одна компонента, крім $x_1 \dots x_m$, несправна, то, згідно властивостям базової моделі K_n^m , в кільцевому графі зникнуть два ребра, і граф втратить зв'язність незалежно від наших перетворень.

Оскільки можлива трансформація любой функції, зрозуміло, що слідє вибирати функцію f_j ($i \neq j$) таким чином, щоб перетворена функція f^* була мінімальною по складності.

Якщо поява вектору станів α не приведе до нуля ні одну з реберних функцій, достатньо таким же образом перетворити дві функції f_i і f_j ($i \neq j$), вибір яких здійснюється по тому ж критерію: мінімальна складність перетворених функцій.

Описана процедура перетворена функцій не міняє функціонування моделі ні на одному векторі, крім α . Тому, якщо втрата зв'язності повинна бути забезпечена на деякій множині векторів, то для кожного з них повинна бути виконана точно така ж процедура.

4.2.2 Додаткові ребра.

Нижчий степінь відмовостійкості системи (по відношенню до базової) може бути визначений на моделі шляхом введення додаткової вершини і ребра, що з'єднує цю вершину з якою-небудь вершиною базової моделі K_n^m . Цьому ребру слідє приписати функцію, рівну одиниці на всіх наборах-векторах станів системи з розглянутої множини (наприклад, представлену у вигляді ДНФ). Можна те ж саме представити у вигляді ланцюга вершин і ребер, кожен з яких має в якості функції ті ж конституенти одиниці. Проте такий варіант гірше, оскільки виключає можливість мінімізації реберних функцій.

В заключенні відмітимо, що оскільки запропоновані в обох розділах перетворення, кожне в окремоті, змінюють вихідну GL-модель лише в рамках своєї цільової функції, вони можуть бути використані незалежно влюбій комбінації.

Розглянемо конкретний приклад.

Нехай існує ВБС з шести модулів і її GL-модель – базова GL-модель K_6^2 . Система була модернізована, в результаті чого вона отримала властивість продовжувати роботу при відмовах наступних груп її модулів:

$$\{1,2,3\}, \{1,2,4\}, \{1,2,5\}, \{1,2,6\}, \{4,2,3\}, \{4,2,5\}, \{4,3,6\}.$$

(4.1)

Крім того, вона втратила властивість бути працездатною при відмові наступних комбінацій її модулів:

$$\{3,5\}, \{3,6\}, \{5,6\}.$$

(4.2)

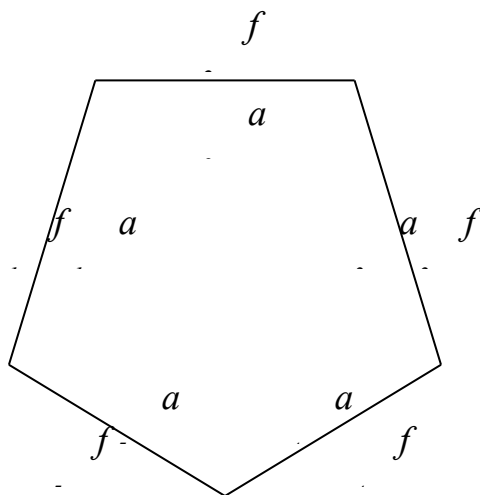
Потрібно побудувати GL-модель модернізованої системи.

Спочатку будуюмо базову GL-модель K_6^2 (см. рис. 4.2).

Послідовно аналізуючи поведінку моделі на всіх векторах станів множини (4.1), встановлюємо пари ребер, які зникають при їх появі

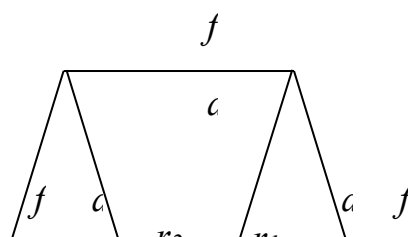
$$(a_1 a_4), (a_1 a_2), (a_1 a_2), (a_1 a_2), (a_1 a_2), (a_2 a_3) \text{ и } (a_2 a_3).$$

Неважко встановити, що для того, щоб виконати основну умову проведення додаткових ребер, достатньо всього двох ребер r_1 і r_2 (рис. 4.3).



$$\begin{aligned} f_1 &= x_1 (x_2 \vee x_3) \vee x_2 x_3 \\ f_2 &= x_4 \vee x_5 \vee x_6 \\ f_3 &= (x_1 \vee x_2 \vee x_3) (x_4 (x_5 \vee x_6) \vee x_5 x_6) \\ f_4 &= x_1 x_2 x_3 \\ f_5 &= x_4 x_5 x_6 \end{aligned}$$

Рис. 4.2



Після складання таблиць істинності функцій φ_1 і φ_2 , відповідним ребрам r_1 і r_2 , з врахуванням наборів, на яких ці функції не визначені, і спрощення форм представлення, отримуємо:

$$\varphi_l = \overline{(x_1 \cup x_2 \cup x_3)(x_4(x_5 \cup x_6) \cup x_5 x_6)}$$

$$\varphi_2 = \overline{(x_1(x_2 \cup x_3) \cup x_2 x_3)(x_4 \cup x_5 \cup x_6)} \cdot \overline{x_1 x_2 x_3}$$

Проаналізуємо тепер поведінку базової моделі при появі векторів станів системи, що відповідають множині (4.2). Можна ввести додаткову вершину і ребро r_3 з функцією φ_3 :

$$\varphi_3 = x_3x_5 \cup x_3x_6 \cup x_5x_6 = x_3(x_5 \cup x_6) \cup x_5x_6$$

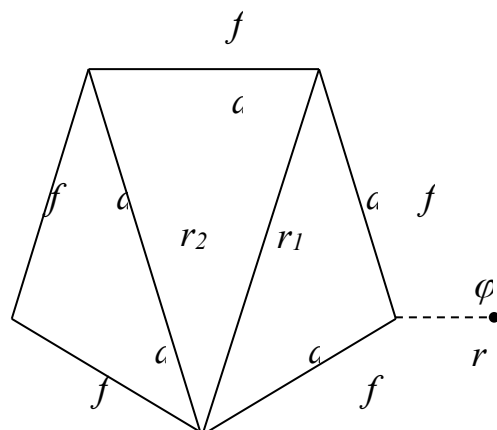


Рис. 4.4

Якщо проаналізувати поведінку моделі на всіх векторах станів системи, можна впевнитись в її адекватності поведінки системи (а це слідує з методики побудови моделі, викладеної вище). Підкреслю ще раз, що операції трансформації моделі - зміни функцій на ребрах і введення додаткових ребер з своїми функціями - проводились незалежно.

4.3 Перетворення МВР-моделей

Як і раніше, будемо розглядати конкретну базову GL-модель K_n^m , представлену у виді циклічного графу R . Позначимо кількість ребер графа як r , а множину його ребер як $A=\{a_1, a_2, a_3 \dots a_r\}$.

Визначимо $W=\{w_1, w_2, w_3 \dots w_n\}$, як множину визначених векторів (наборів) станів модулів (вхідних змінних $x_1, x_2 \dots x_n$), в яких кратність відмови перевищує m , але в той же час система продовжує функціонувати (блокування векторів).

Задача оптимізації полягає в побудові розширеної GL-моделі (згідно 2.2), що відображає поведінку ВБС на множині W з використанням мінімально можливої кількості додаткових ребер.

Визначимо величину t_i як кількість ребер графу R , зникаючих на наборі w_i , а також максимальну кількість одночасно зникаючих ребер $t_{max}=\max(t_i)$.

Будемо вирішувати задачу оптимізації шляхом побудови графу R^* на основі R , впорядковуючи існування і проводячи додаткові ребра для забезпечення збереження зв'язності графу R^* при появі любого вектора з W (далі просто «забезпечення зв'язності»).

Кожний набір з W буде визначати сукупність ребер графу R , на яких функції приймають нульове значення. Множину таких сукупностей ребер позначимо $Q=\{q_1, q_2, \dots q_n\}$, $q_k=\{a_{1k}, a_{2k}, \dots\}$.

Позначимо множину всіх пар ребер, що входять в q_i , як q_i^* .

Побудуємо множину $P = q'_1 * q'_2 * \dots * q'_n$, тобто множину, що складається з всеможливих пар ребер, що входять в елементи множини Q .

Очевидно, що для випадку $t_{max}=2$ $P=Q$. Також зрозуміло, що при $t_{max}=3$, в множині P присутні всі елементи Q , що складаються з 2-х ребер, а всі елементи Q , що складаються з 3-х ребер, представлені трьома парами ребер.

Твердження 4.1: для забезпечення збереженості зв'язності графу при зникненні основних ребер із заданої множини q_i ($|q_i| > 2$) шляхом проведення додаткових ребер необхідно забезпечити збереження зв'язності графу при зникненні кожної пари з q_i^* .

Примітка: достатність, в загальному випадку, не утверджується.

В якості доведення, достатньо відмітити, що, якщо зникнення хоча б однієї пари ребер з q_i^* приведе до втрати зв'язності графу, то при зникненні всіх ребер з q_i вона тим більше буде втрачена.

Далі, нехай є дві GL-моделі G_1 і G_2 з однаковими графами і однаковими множинами P . Значення t_{max} у моделях позначимо t^1_{max} і t^2_{max} , а необхідна і достатня для забезпечення зв'язності кількість додаткових ребер, відповідно, як e_1 і e_2 .

Твердження 4.2: з $t^1_{max} < t^2_{max}$ слідує $e_1 \leq e_2$.

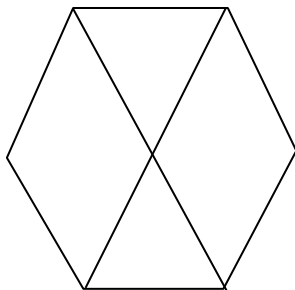
Доведення: Допустимо зворотнє, а саме, що існують такі моделі G_1 і G_2 з $t^1_{max} < t^2_{max}$, для яких $e_1 > e_2$. згідно твердженню 1, що застосовується до кожного елементу множини Q моделі G_2 , e_1 є необхідною кількістю додаткових ребер для збереження зв'язності графу моделі G_2 . Виходить, що необхідне число для однієї і тої ж моделі більше достатнього ($e_1 > e_2$). Останнє неможливе, а отже вихідне твердження справедливе.

Наслідок: якщо $t^1_{max} < t^2_{max}$ і в моделі G_2 проведено деяка достатн' для забезпечення зв'язності кількість ребер e^* , така що $e^* = e_1$, то $e_2 = e^*$, тобто ця ж кількість ребер буде і необхідною.

S-підмножини. Нехай є граф R^* , в якому проведено декілька додаткових ребер з деякими функціями, приписаними їм.

Два або більше непустих підмножин основних ребер, що не перетинаються, визначимо як S-підмножини, якщо зв'язність графу зберігається при втраті будь-якої пари ребер, що належать різним підмножинам, і якщо зв'язність графу порушується при втраті будь-якої пари ребер, що належать будь-якій з підмножин, що містить більше одного ребра.

Розглянемо приклад: Додаткове ребро (для простоти припускаємо, що його функція $f=I$) розділяє множину основних ребер A на дві підмножини A_1, A_2 , що не перетинаються і мають наступну властивість: зв'язність графу зберігається при втраті будь-якої пари ребер $\{a_i, a_j\}$ $a_i \in A_1, a_j \in A_2$. З іншої сторони, зв'язність графу порушиться при втраті будь-якої пари з A_1 або з A_2 . В даному випадку A_1 і A_2 є S-підмножинами.



Кількість ребер циклічного графу: $r=6$

Кількість S-підмножин: $\lambda=4$

S-підмножини: $S=\{\{a_1\}, \{a_2, a_3\}, \{a_4\}, \{a_5, a_6\}\}$

Кількість додаткових ребер: $e=2$

Кількість «захищених» пар: $\{a_1, a_2\}, \{a_1, a_3\}, \{a_1, a_4\}, \{a_1, a_5\}, \{a_1, a_6\}, \{a_2, a_4\}, \{a_3, a_4\}, \{a_2, a_5\}, \{a_2, a_6\}, \{a_3, a_4\}, \{a_3, a_5\}, \{a_3, a_6\}, \{a_4, a_5\}, \{a_4, a_6\}, \{a_5, a_6\}$

Рис

Кожне нововведене додаткове ребро розділяє вже отримані підмножини на більше дрібні. Таким чином, можливе розбиття множини A на довільну кількість S-підмножин, що не перевищує кількість основних ребер. По S-підмножинам легко отримувати множину «захищених» пар ребер (під «захищеною» парою розуміємо пару основних ребер, при зникненні яких зв'язність графу зберігається).

На рис. 4.5 приведено приклад графу розширеної GL-моделі з двома додатковими ребрами, що розділяють множину основних ребер на 4 S -підмножини.

З сказаного стає зрозуміло, що для забезпечення зв'язності на заданих наборах при $t_{max}=2$ можна побудувати граф R^* , впорядкувавши ребра графу R і провівши додаткові ребра, тим самим розділивши основні на S -підмножини так, що вершини кожної з пар (p_i) знаходились в різних підмножинах. Функції, приписані додатковим ребрам, повинні формуватись таким чином, щоб зв'язність графу, при появі інших наборів, залишалась відповідною до базової моделі.

Модель, що втрачає мінімум ребер (тобто $t_{max}=2$) при появі однієї зайвої відмови називається **МВР-моделлю**.

Мінімальна кількість S -підмножин в розширеній GL-моделі для випадку $t_{max}=2$ (зведення задачі до розфарбування графу).

Спочатку вирішимо задачу про необхідну і достатню кількість S -підмножин для забезпечення зв'язності графу при зникненні якої-небудь пари ребер з множини P .

Для цього побудуємо граф V : $V=\{A,P\}$. Іншими словами, в якості множини вершин візьмемо множину ребер графу R : $A=\{a_1, a_2, a_3 \dots a_R\}$, а в якості ребер - множину пар P . Це означає, що вершини a_i і a_j в графі V будуть суміжними, якщо необхідно забезпечити зв'язність графу R при відсутності ребер a_i і a_j .

Очевидно, що визначення S -підмножин ребер графу R еквівалентно визначенню внутрішньо стійких множин вершин графу V (внутрішньо стійкою множиною вершин називається підмножина вершин графу не зв'язних між собою [2]). Таким чином, знайшовши розбиття множини вершин графу V на мінімальне число внутрішньо стійких множин, ми знаходимо шукані S -підмножини ребер графу R .

Задача пошуку S -підмножин графу R тепер може бути зведена до розфарбування вершин графу V [2]. Вершини, що отримали один колір, будуть належати одній внутрішньо стійкій множині. Позначимо кількість кольорів, або, що те ж саме, кількість отриманих внутрішньо стійких множин як λ , що не перетинаються. Отримані множини будемо позначати $s_i = \{a_{i1}, a_{i2}, \dots, a_{ij}\}$, $S = \{s_1, s_2, s_3 \dots s_\lambda\}$.

Мінімальне число таких підмножин (λ_{min}) дорівнює хроматичному числу графу V . Кількість варіантів розбиття на підмножини визначається значенням хроматичного поліному [3] графу V по значенню λ .

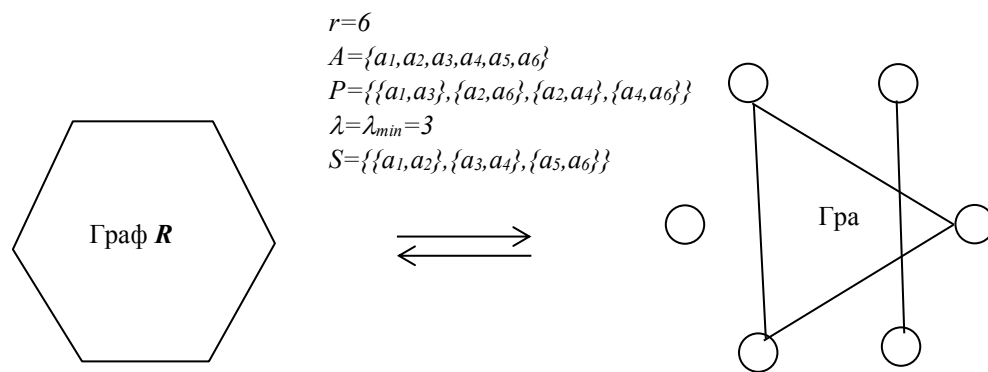


Рис. 4.6

Приклад побудови графу V і його розфарбування.

Евристичний алгоритм розфарбування графу, що не дає в загальному випадку точного розфарбування [3], представляє для нас певну цінність: якщо з допомогою евристичного алгоритму розфарбування було отримано розбиття з $\lambda \leq 3$ то $\lambda_{min} = \lambda$. В такому випадку немає необхідності приміняти алгоритм точного розфарбування, що може скоротити час обрахунків. З іншої сторони, для графу V з $\lambda_{min} \leq 2$ алгоритм завжди дасть $\lambda = \lambda_{min}$. Доведення ґрунтується на тому, що хроматичне число не перевищує 2 тоді і тільки тоді, коли в графі немає непарних циклів. Іншими

словами, якщо достатньо одного додаткового ребра для переходу від графу R до R^* (тобто від базової моделі к розширеної), то евристичний алгоритм завжди дасть правильний результат.

Оптимальна кількість додаткових ребер для $t_{max}=2$. Розфарбування графу V дозволяє визначити мінімальну кількість (λ_{min}) S -підмножин, а також знайти самі S -підмножини.

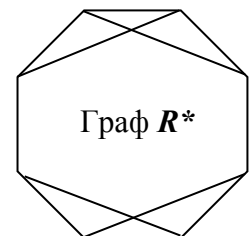
Вважаючи, що $\lambda = \lambda_{min}$, позначимо необхіду і достатню кількість додаткових ребер для отримання λ S -підмножин як e .

Твердження 4.3: $e = \lceil (\lambda + 1) / 2 \rceil$.

Доведення. Для початку розглянемо найгірний варіант: $\lambda = r$ (кількість S -підмножин рівна кількості ребер графу R). Цей варіант має місце в тому випадку, коли необхідно забезпечити зв'язність графу при зникненні будь-якої пари ребер. В цьому випадку всі можливі пари ребер будуть перераховані в множині P , тобто граф V , буде повним. Дійсно, розфарбування повного графу V дає хроматичне число $\lambda_{min} = r$.

Очевидно, що для збереження зв'язності графу R^* при втраті будь-яких 2-х ребер необхідно, щоб у кожній його вершини було як мінімум 3 ребра (1 додаткове). Оскільки кожне додаткове ребро з'єднує 2 вершини графу R^* , необхідна кількість ребер $e|_{\lambda=r} = \lceil (\lambda + 1) / 2 \rceil$, де $\lceil x \rceil$ – ціла частина x .

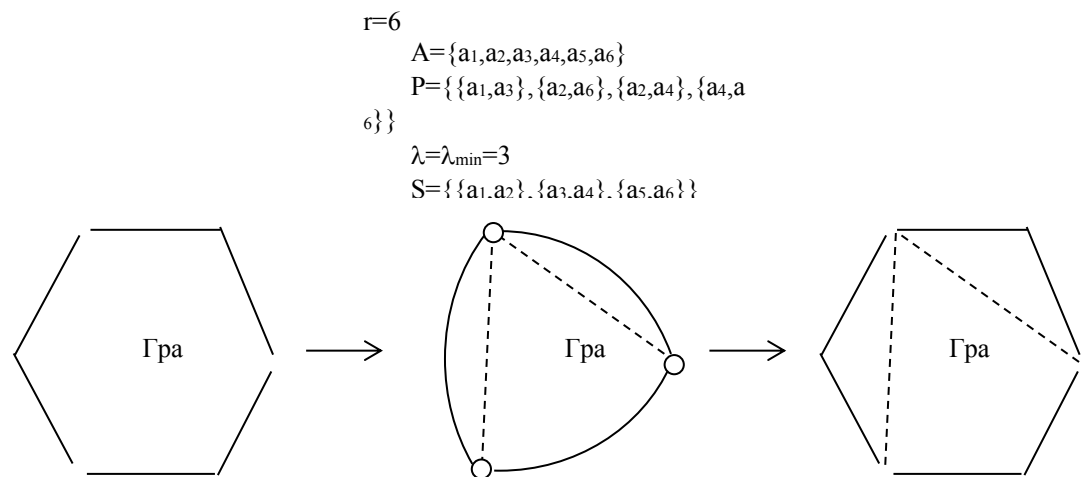
В той же час при $r > 4$ не кожен варіант з'єднання всіх вершин графу R^* $e|_{\lambda=r}$ ребрами дасть розбиття на λ S -підмножин (див., наприклад, рис. 4.7).



Рис

Все додаткові ребра можуть бути проведені так, щоб кожне додаткове ребро збільшувало на два кількість підмножин ребер графу R^* (за винятком останнього ребра при непарному λ), що і визначає алгоритм побудови додаткових ребер. Отже, необхідна і достатня кількість додаткових ребер для окремого (найгіршого) випадку $e|_{\lambda=r} = \lceil (\lambda+1)/2 \rceil$. Тепер розглянемо загальний випадок, в якому кількість S -підмножин може бути довільною. Очевидно, що кожне нововведенне додаткове ребро може збільшувати кількість S -підмножин ребер не більше ніж на два. З цього слідує, що необхідне число ребер для розбиття на λ S -підмножин: $e = \lceil (\lambda+1)/2 \rceil$. Користуючись тою властивістю, що базова GL-модель не накладає обмеження на взаємне розташування основних ребер, впорядкуємо ребра графу R так, щоб ребра, що належать одній підмножині, утворювали ланцюг.

Рис. 4.8



Побудуємо циклічний граф S , що складається з λ ребер, що відповідають S -підмножинам $S = \{s_1, s_2, s_3 \dots s_\lambda\}$. Іншими словами, перетворимо граф R шляхом заміни кожного ланцюгу одним ребром у

відношенні з S -підмножиною ребер, визначених на етапі розфарбування графу V . Заміна ланцюгу одним ребром не обмежує загальності, так як в одній S -підмножині не може одночасно зникнути більше одного ребра при будь-якому наборі з W (по побудові S -підмножин).

Так як при переході до графу S ми отримали граф, в якому кількість S -підмножин і ребер співпадає, загальний випадок можна звести до розглянутого вище. Додаткові ребра можна проводити, модифікуючи безпосередньо граф S , так як можливий зворотній перехід, шляхом заміни ребер графу S вихідними ланцюгами (див., наприклад, рис. 4.8).

Таким чином, необхідною і достатньою кількістю додаткових ребер для отримання λ S -підмножин є $e = \lceil (\lambda + 1)/2 \rceil$, що і треба було довести.

Твердження 4.4: люба конфігурація додаткових ребер, що забезпечує збереження зв'язності графу S при зникненні любых двох основних ребер, забезпечує збереження зв'язності графу при зникненні любых трьох основних ребер.

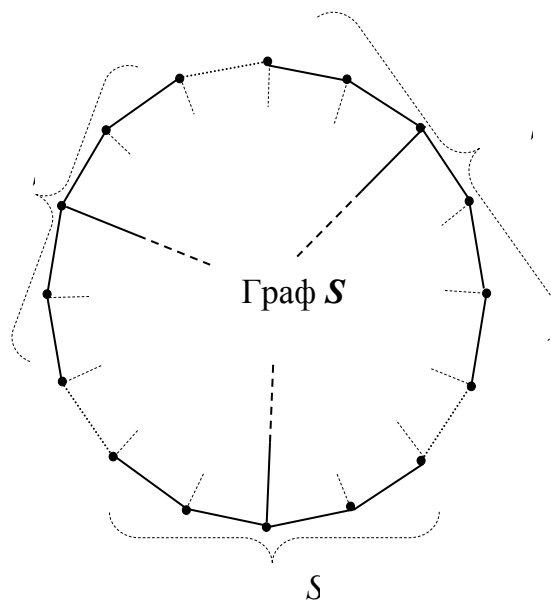


Рис.

Для доведення розглянемо довільний граф S з певною конфігурацією додаткових ребер, що зберігає зв'язність при зникненні будь-якої пари основних. Розглянемо довільні три ребра s_i, s_j, s_k .

Ребра s_i, s_j, s_k при видаленні з графу S (рис. 4.9) розділяють підграф, створений основними ребрами, на три зв'язних підграфа. Підграф, що з'єднує ребра s_i і s_j , позначимо як S_k ; об'єднуючий s_i і s_k позначимо як S_j ; об'єднуючий s_j і s_k позначимо як S_i .

Згідно вихідному положенні твердження, граф S зберігає зв'язність при втраті будь-якої пари ребер, в тому числі: $(s_i, s_j), (s_j, s_k), (s_i, s_k)$.

Розглянемо підграф S_k . В цьому підграфі є як мінімум одна вершина, інцидентна додатковому ребру, що з'єднує даний підграф з підграфом S_i чи з S_j . В іншому випадку граф втратив би зв'язність при зникненні пари (s_i, s_j) .

Аналогічні твердження справедливі для підграфів S_i і S_j , тобто в підграфі, що об'єднує відповідні ребра, в парі є як мінімум одна вершина, інцидентна додатковому ребру, що об'єднує даний підграф з одним із тих двох, що залишились.

Таким чином, в графі S є ребра, що з'єднують три розглянутих підграфа в одну компоненту зв'язності, що містить всі вершини графу S . З цього слідує, що граф S збереже зв'язність при втраті ребер s_i, s_j, s_k . Ребра були вибрані довільно, тому граф S зберігає зв'язність при втраті будь-яких трьох ребер. Що і треба було довести.

Розглянемо GL-модель з $t_{max}=3$, тобто ту, в якій множина Q містить хоча б одну трійку ребер.

Граф S для випадку $t_{max}=2$ був побудований таким чином, щоб при проведенні мінімальної кількості додаткових ребер и подальшому зворотньому переході до графу R забезпечувати його зв'язність при втраті будь-якої пари ребер з множини P , тобто при втраті будь-якої пари основних ребер S .

Згідно твердженню 4.4, граф S збереже зв'язність при втраті любых трьох основних ребер. Граф R^* при переході до нього від графу S збереже зв'язність при втраті пар і трійок ребер з множини Q , з чого слідує, что описаний в 4.2 і 4.3 спосіб проведення додаткових ребер дає достатню кількість ребер для випадку $t_{max}=3$. Згідно наслідку з твердження 4.2, ця кількість ребер буде і необхідною, а, отже, оптимальною.

Відповідно, твердження 4.1 можна уточнити для окремого випадку:

Твердження 4.1а: для забезпечення збереження зв'язності графу при зникненні заданої трійки основних ребер шляхом проведення додаткових ребер, необхідно і *достатньо* забезпечити збереження зв'язності графу при зникненні кожної пари ребер з цієї трійки.

Заради справедливості варто відмітити, що прямого узагальнення приведенного підходу для випадку $t_{max} \geq 4$ не існує. Дослідження таких складніших и більш загальних випадків буде предметом подальшої діяльності.

4.4 Попарні реберні цикли при перетворенні GL-моделей

При оптимізації перетворення моделей шляхом проведення внутрішніх ребер можуть виникнути певні труднощі. Однією з причин цього є так звані попарні реберні цикли (ПРЦ), суть яких полягає в наступному. Нехай необхідно блокувати деяку множину векторів стану системи, серед яких є вектори w_1 , w_2 і w_3 , кожен з яких викликає зникнення своєї пари ребер. Якщо це пари r_1, r_2 ; r_2, r_3 та r_3, r_1 відповідно, то вони утворюють своєрідний цикл – ПРЦ. Такий цикл може містити в собі будь-яку кількість ребер. Однак парний цикл для блокування всіх векторів, що приводять до появи ПРЦ, потребує всього одного додаткового ребра, тоді як непарний - потребує більше. Саме ця особливість ПРЦ змушує нас досліджувати це явище докладніше.

Існують методи зменшення кількості ПРЦ в GL -моделі аж до їх ліквідації. Наприклад, можна змінити хоча б одну реберну функцію так, як зазначено вище. При використанні подібних методів важливо знати кількість ПРЦ, що виникають у моделі. Якщо кількість векторів стану ВБС, які необхідно блокувати, не велике, то визначити кількість ПРЦ нескладно. Однак, при значних кількостях векторів, перебір стає занадто складним.

Виникає задача встановлення умов існування ПРЦ, вирішення якої спростило б пошук ПРЦ в GL -моделях і тим самим скоротило б згаданий перебір. В даній роботі ми обмежимося випадком $K(3,n)$ і блокуванням векторів стану системи, що містять 4 нульові компоненти (будемо позначати множину таких векторів як W), а також обмежимося розглядом створення ПРЦ тільки трьома парами ребер, хоча все сказане нижче може бути узагальнене для будь-якого n .

Нехай є три вектори $w_i \in W$, кожний з яких викликає зникнення в моделі однієї пари ребер. Спочатку сформулюємо майже очевидне твердження.

Твердження 4.1.

ПРЦ має місце, якщо і тільки якщо:

- 1) число різних ребер, що містяться в цих трьох парах, дорівнює 3;*
- 2) жодна пара не повторюється.*

Наслідок 4.1.

Якщо якась пара, наприклад, (r_1, r_2) , не випадає, то ПРЦ не має місця.

Дійсно, у цьому випадку будь-яка пара ребер випадає з появою і w_k , і w_l ($k, l = 1, 2, 3, k \neq l$) і, отже, ПРЦ неможливий.

Твердження 4.1 і його наслідок дозволяють спростити процедуру визначення існування ПРЦ для заданої трійки ребер, які випадають в базовій моделі. Однак при більших кількостях векторів, які необхідно блокувати, перебір залишається значним. Тому визначення додаткових умов існування ПРЦ, які сприяли б зменшенню цього перебору, становить

інтерес. Спочатку зупинимося на умовах, при яких може випадати будь-яка пара ребер. Для цього повернемося до дерева ієрархії реберних функцій, описаному в підрозділі 3.2.

Твердження 4.2.

Два ребра GL -моделі базової ВБС $K(3,n)$ можуть одночасно випадати з появою вектора стану $w_i \in W$ тоді і тільки тоді, коли їх реберні функції відповідають двом вершинам дерева ієрархії, які:

- *або розташовані на одній гілці дерева;*
- *або розташовані на одному рівні ієрархії дерева і мають загальну суміжну вершину.*

Для доведення необхідності обох умов сформулюємо порівняно прості Лема 4.1 і 4.2.

Лема 4.1.

Не існує вектор $w_i \in W$, в якому нульові компоненти розташовані таким чином, щоб в GL -моделі $K(3,n)$ випадали 2 ребра, які відповідають вершинам дерева одного рівня ієрархії ті не мають загальної суміжної вершини.

Прикладом, що ілюструє розташування вершин, можуть служити вершини 112 і 121 дерева (див. рис. 3.1 з параграфа 3.2).

Справді, виходячи зі способу побудови GL -моделі, можна сказати, що кожна функція, яка приписана ребру, що відповідає вершині дерева, приймає значення 0 лише коли принаймні три змінні, що належать певній підмножині, дорівнюють 0. З іншого боку, спосіб побудови дерева показує, що для двох вершин, які розташовані на одному рівні дерева ієрархії та не мають загальної суміжної вершини, ці підмножини не перетинаються. Оскільки вектор $w_i \in W$ містить всього 4 нульові компоненти, лему можна вважати доведеною.

Лема 4.2.

Не існує вектор $w_i \in W$, в якому нульові компоненти розташовані таким чином, щоб в GL -моделі $K(3,n)$ випадали 2 ребра, функції яких відповідають вершинам дерева, розташованим на різних рівнях різних гілок дерева ієрархії.

Доведення аналогічно.

Повернемося до доведення *твердження* 4.2. Зовсім інші міркування мають місце в тому випадку, якщо вершини, розташовані на одному рівні дерева ієрархії, мають загальну інцидентну вершину. Останнє означає, що для того, щоб реберні функції, що відповідають розглянутим вершинам, приймали нульове значення, нульові компоненти вектора $w_i \in W$ повинні бути присутніми в одній підмножині, розбитій у свою чергу на дві підмножини, в кожній з яких повинно бути по дві нульові компоненти. Сказане, мабуть, можливо, що і обумовлює достатність умови б).

Наприклад, для вершин 121 і 122 поява вектора з нульовими компонентами на місцях 8, 9, 14, 15 призведе до обнуління функцій, приписаних відповідним ребрам.

Перейдемо до завершення доведення – достатності умови а) *твердження*. Той факт, що 2 вершини лежать на одній гілці дерева означає, що функція, яка відповідає вершині, більш віддаленої від кореня дерева, залежить від деякої підмножини множин змінних функції, що відповідає іншій вершині. Це впливає зі способу побудови дерева, з одного боку, а з іншого боку – це означає, що потрібний вектор існує. Справді, три нульових компоненти вектора розташовані на місцях змінних, які визначають нульове значення функції f_i , що відповідає вершині, більш віддаленої від кореня. При цьому змінні f_j , про які йде мова, входять в одну з двох підмножин змінних функції, що відповідає вершині, яка розташована ближче до кореня, а саме в підмножину, що утворює $\sigma_2(2, a_i)$. Четвертий нульовий компонент розташована в іншій підмножині множини

змінних функції f_j , тобто підмножини змінних, розташованих з іншої сторони знака диз'юнкції цієї функції та утворюючого $\sigma_l(1, a_j)$.

Наприклад, вершинам 1 та 1122 відповідає вектор стану ВБС з нульовими компонентами на місцях 5, 7, 8, 16. Звичайно, він не єдиний.

Доведення завершене.

Визначимо кількість векторів, які призводять до зникнення пари ребер, використовуючи для цього те ж дерево ієрархії. Реберні функції $K(3,31)$ моделі (кожна з яких відповідає деякій вершині дерева), отримані за способом формування МВР-моделі, наступні:

$$f_1 = x_1 \vee x_2 \vee x_3$$

$$f_2 = (x_1 \vee x_2)(x_1 x_2 \vee x_3) \vee x_4 x_5 x_6 x_7$$

$$f_3 = x_1 x_2 x_3 \vee (x_4 \vee x_5)(x_6 \vee x_7)(x_4 x_5 \vee x_6 x_7)$$

$$f_4 = x_4 x_5 \vee x_6 \vee x_7$$

$$f_5 = x_4 \vee x_5 \vee x_6 x_7$$

$$f_6 = x_8 x_9 \vee x_{10} \vee x_{11}$$

$$f_7 = x_8 \vee x_9 \vee x_{10} x_{11}$$

$$f_8 = (x_8 \vee x_9)(x_8 x_9 \vee x_{10} x_{11})(x_{10} \vee x_{11}) \vee x_{12} x_{13} x_{14} x_{15}$$

$$f_9 = x_8 x_9 x_{10} x_{11} \vee (x_{12} \vee x_{13})(x_{12} x_{13} \vee x_{14} x_{15})(x_{14} \vee x_{15})$$

$$f_{10} = x_{12} x_{13} \vee x_{14} \vee x_{15}$$

$$f_{11} = x_{12} \vee x_{13} \vee x_{14} x_{15}$$

$$f_{12} = \overbrace{(x_1 \vee x_2)(x_1 x_2 \vee x_3)(x_1 x_2 x_3 \vee x_4 x_5 x_6 x_7)(x_4 \vee x_5)(x_6 \vee x_7)(x_4 x_5 \vee x_6 x_7)}^{\sigma_1} \vee x_8 x_9 x_{10} x_{11} x_{12} x_{13} x_{14} x_{15}$$

$$f_{13} = \overbrace{(x_8 \vee x_9)(x_{10} \vee x_{11})(x_8 x_9 \vee x_{10} x_{11})(x_{12} \vee x_{13})(x_{14} \vee x_{15})(x_{14} x_{15} \vee x_{12} x_{13})(x_8 x_9 x_{10} x_{11} \vee x_{12} x_{13} x_{14} x_{15})}^{\sigma_2} \vee x_1 x_2 x_3 x_4 x_5 x_6 x_7$$

$$f_{14} = [\sigma_1 \cdot \sigma_2 \cdot (x_1 x_2 x_3 x_4 x_5 x_6 x_7 \vee x_8 x_9 x_{10} x_{11} x_{12} x_{13} x_{14} x_{15})] \vee x_{16} x_{17} x_{18} \dots x_{31}$$

$$f_{15} = \overbrace{(x_{16} \vee x_{17})(x_{18} \vee x_{19})(x_{16}x_{17} \vee x_{18}x_{19})}^{\sigma_3} \vee x_{20}x_{21}x_{22}x_{23}$$

$$f_{16} = \overbrace{(x_{20} \vee x_{21})(x_{22} \vee x_{23})(x_{20}x_{21} \vee x_{22}x_{23})}^{\sigma_4} \vee x_{16}x_{17}x_{18}x_{19}$$

$$f_{17} = x_{20}x_{21} \vee x_{22} \vee x_{23}$$

$$f_{18} = x_{20} \vee x_{21} \vee x_{22}x_{23}$$

$$f_{19} = x_{16} \vee x_{17} \vee x_{18}x_{19}$$

$$f_{20} = x_{16}x_{17} \vee x_{18} \vee x_{19}$$

$$f_{21} = \overbrace{(x_{24} \vee x_{25})(x_{26} \vee x_{27})(x_{24}x_{25} \vee x_{26}x_{27})}^{\sigma_5} \vee x_{28}x_{29}x_{30}x_{31}$$

$$f_{22} = \overbrace{(x_{28} \vee x_{29})(x_{30} \vee x_{31})(x_{28}x_{29} \vee x_{30}x_{31})}^{\sigma_6} \vee x_{24}x_{25}x_{26}x_{27}$$

$$f_{23} = x_{28} \vee x_{29} \vee x_{30}x_{31}$$

$$f_{24} = x_{28}x_{29} \vee x_{30} \vee x_{31}$$

$$f_{25} = x_{24} \vee x_{25} \vee x_{26}x_{27}$$

$$f_{26} = x_{24}x_{25} \vee x_{26} \vee x_{27}$$

$$f_{27} = \overbrace{\sigma_3 \cdot \sigma_4 \cdot (x_{16}x_{17}x_{18}x_{19} \vee x_{20}x_{21}x_{22}x_{23})}^{\delta_1} \vee x_{24}x_{25} \dots x_{31}$$

$$f_{28} = \overbrace{\sigma_5 \cdot \sigma_6 \cdot (x_{24}x_{25}x_{26}x_{27} \vee x_{28}x_{29}x_{30}x_{31})}^{\delta_2} \vee x_{16}x_{17} \dots x_{23}$$

$$f_{29} = \delta_1 \cdot \delta_2 \cdot (x_{16}x_{17} \dots x_{23} \vee x_{24}x_{25} \dots x_{31}) \vee x_1x_2x_3 \dots x_{15}$$

Звернемося до *твердження 4.2* і розглянемо умову *a)*, тобто випадок з двома вершинами одного рівня. Якщо пригадати, що функції, які приписуються ребрам, що відповідають цим вершинам, представляються виразами:

$$f_{i1} = \varphi_1(2, n_1) \vee \varphi_2(1, n_2),$$

$$f_{i2} = \psi_1(1, n_1) \vee \psi_2(2, n_2),$$

то стане зрозумілим, що кількість векторів з множини W , які обнуляють цю пару функцій, дорівнюють

$$S_{ii} = C_{n1}^2 \times C_{n2}^2,$$

оскільки кожна з них повинна містити по дві нульові компоненти в кожній з підмножин n_1 та n_2 змінних. Для простоти в виразі для S_{ii} (і далі також для S_{ij}) приймаємо одне й те саме позначення – n_i – для множини n_i та його потужності. Наприклад, ребра, що відповідають вершинам дерева 11 і 12, випадають одночасно з появою кожного з

$$C_7^2 \times C_8^2 = 588$$

векторів. У той же час зникненню ребер, що відповідають вершинам 1121 і 1122, відповідає єдиний вектор

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	...	x_{31}
1	1	1	0	0	0	0	1	...	1

З іншого боку, пара ребер, що відповідають вершинам, розташованим на одній гілці дерева, тобто умові а) з *твердження 4.2*, з функціями f_i (ближче до кореня) і f_j

$$f_i = \varphi_1(2, n_{i1}) \vee \varphi_2(1, n_{i2}),$$

$$f_j = \psi_1(2, n_{j1}) \vee \psi_2(1, n_{j2})$$

випадає в *GL*-моделі з появою кожного з

$$S_{ij} = C_{ni2}^1 \times C_{nj1}^2 \times C_{nj2}^1$$

векторів. Дійсно, нагадаємо, що, відповідно до принципу побудови дерева, підмножини змінних n_{j1} та n_{j2} належать множині n_{i1} , і нульові компоненти вектора в підмножині n_{i1} розташовані в строгій відповідності: дві – в n_{j1} і одна – в n_{j2} . Четвертий нульовий компонент розташований у підмножині n_{i2} , що в сукупності пояснює вираз для S_{ij} .

Перейдемо до формулювання умов появи ПРЦ.

Твердження 4.3.

Для трійки ребер GL -моделі та відповідних їм трьом вершинам a , b , c дерева ієрархії ПРЦ можливий тоді і тільки тоді, коли:

1) одна з вершин дерева (скажімо, вершина “ a ”) лежить ближче до двох інших до кореня дерева, і

2.1) або існує ланцюжок на дереві, на якому лежать всі 3 вершини,

2.2) або вершини b і c є суміжними з вершини d , і вершини a і d лежать на одній гілці.

Доведення почнемо з необхідності.

Досліджуємо можливі варіанти взаємного розташування вершин, про які йде мова в твердженні.

Випадок, коли всі три вершини лежать на одному рівні, ми виключаємо з розгляду, оскільки на нашому дереві не можуть три вершини бути суміжними з однією, і, отже, якась пара з трійки ребер, що відповідають розглянутим вершинам, не може випадати одночасно відповідно до умов твердження 4.2, і це означає, що ПРЦ неможливо.

Далі, нехай будь-які дві вершини лежать на одному рівні й одна – нижче. Цей випадок ми теж виключаємо як неможливий, оскільки ребра хоча б однієї з пар не можуть випадати одночасно (за умовою *а твердження 4.2*), оскільки відповідні вершини лежать на різних гілках дерева.

Ще один варіант – всі вершини лежать на різних рівнях, причому хоча б дві з них лежать на різних гілках дерева. Цей випадок також виключаємо, оскільки саме ця пара вершин вказує на ту пару ребер, які не можуть випадати одночасно відповідно до умови *а твердження 4.2*.

Інші варіанти підпадають під умови твердження. Необхідність умови 1) можна вважати доведеною.

Необхідність умов 2.1 та 2.2 *твердження 4.3* впливає безпосередньо з *лем 4.1 та 4.2*, оскільки будь-яке інше взаємне

розташування вершин говорить про те, що для деякої пари з них не існує вектора, що спричинив би пропадання в моделі відповідної пари ребер.

Перейдемо до достатності. Це зробити нескладно. Справді, якщо всі 3 вершини a , b і c задовольняють умовам 1 і 2.1 твердження, то ПРЦ можливий, оскільки, відповідно до *твердження 4.2*, існують 3 вектори $w_i \in W$, поява яких викличе пропадання трьох пар ребер, які відповідають парам вершин ab , bc і ac .

Інший варіант, якщо розташування вершин на дереві ієрархії задовольняє умовам 1 і 2.2, то ПРЦ також можливий. Нехай вершини b і c це - ті, які мають загальну суміжну вершину (скажемо d). Відповідно до *твердження 4.2*, вектор, що викликає пропадання в GL -моделі пари відповідних їм ребер, існує. Вершина a лежить ближче до кореня, і вона лежить на одній гілці з вершиною b і на одній гілці з вершиною c (оскільки вона за умовою 2.2 лежить на одній гілці з вершиною d). Отже, вектори, поява яких приводить до пропадання пар ребер, що відповідають парам вершин ab і ac , теж існують.

Доведення завершено.

Наслідок 4.2.

Ребра, що відповідають двом вершинам одного рівня та найбільш близькі до кореня, не входять ні в який ПРЦ.

Дійсно, умова 1 твердження 4.3 не виконується і, крім того, вершина з номером 0 не має відповідного ребра в GL -моделі.

Етапи знаходження всіх трьохреберних ПРЦ.

1. Із заданої множини W векторів стану ВБС, які необхідно блокувати, виділити множину S ребер і множину P пар ребер, які випадають при появі цих векторів.

2. В множині S виділити множину Q трійок ребер, які задовольняють умовам 4.1 і 4.2 *твердження* 4.3.

3. Для кожного елемента q_i множини Q виділити підмножину R_i елементів p_i множини P , для яких виконується $p_i \in q_i$.

4. У кожній множині R_i виділити підмножини T_i , які задовольняють умовам *твердження* 4.1. Поява відповідних векторів стану ВБС призводить до утворення ПРЦ в моделі.

Питання оптимізації GL -моделей, що відображають реакцію ВБС на появу відмов будь-якої кратності, дуже важливі, зокрема, на етапі перетворення базової моделі в небазову. Оптимізація моделі спрощує модель за деякими параметрами, що дає можливість зменшити час статистичного експерименту і тим самим підвищити точність розрахунку надійності ВБС вже на етапі проектування. Описані вище порівняно прості умови, що дозволяють зменшити перебір при виявленні ПРЦ на основі введеного поняття дерева ієрархії ребер GL -моделі, дають можливість спростити процедуру перетворення базової моделі в першу чергу для ВБС управління складними об'єктами.

4.1 Висновки

В главі розглядаються питання організації перетворення базових моделей поведінки відмовостійких багатопроцесорних систем в потоці відмов. Оскільки реально далеко не всі ВБС є базовими виникає задача перетворення базових моделей до небазових таким чином, щоб адекватність моделі поведінки системи збереглась. Пропонується використовувати для нової моделі два шляхи вирішення цієї задачі: введення додаткових ребер в граф моделі і зміна реберних функцій моделі. Перший шлях передбачає появу нових ребер зі своїми функціями в моделі, які перешкоджають втраті зв'язності графом моделі при появі визначених

векторів стану системи з підмищеною кратністю відмов. Розглядається задача оптимізації проведення додаткових ребер, а саме, їх кількості і місця проведення. Рішення задачі здійснюється на основі розфарбування графів. Другий шлях, з одного боку, простіший, так як не ускладнює вирішення задачі визначення зв'язності графу моделі на кожному векторі стану системи. Проте з іншого боку, іноді сильно ускладнюються самі реберні функції моделі, що може спричинити збільшення часу моделювання.

Аналізується більш детально можливість перетворення моделі шляхом введення додаткових ребер. Показується, що при цьому можуть виникати попарні реберні цикли (ПРЦ), які ускладнюють перетворення, особливо в тих випадках, коли число випадajuчих ребер непарне. Пропонується алгоритм визначення всіх ПРЦ, якщо відома кількість векторів стану системи, яку необхідно блокувати, що іноді корисно знати розробнику системи.

ВИСНОВКИ

Вирішена задача побудови нової GL-моделі стійкої до двох відмов, до довільної кратності відмов, вирішена задача мінімізації отриманих моделей. Відмінною особливістю даної моделі від існуючих є простота формування реберних булевих функцій, можливість запису системи реберних булевих функцій у вигляді однієї функції, рекурсивність алгоритму мінімізації.

Поставлені і вирішені задачі модернізації базових GL-моделей з метою відображення реальної поведінки ВБС при появі визначеного числа відмов, яке відрізняється від допустимого в базових системах, як в більшу, так і в меншу сторони. Запропоновані способи перетворення реберних функцій, причому модель залишається циклічною. Запропоновано також другий спосіб, пов'язаний з проведенням додаткових ребер, який виводить модель з класу циклічних, вирішена задача формування реберних функцій для додаткових ребер. Поставлена і вирішена задача визначення і розміщення оптимального числа додаткових ребер в конкретній базовій GL-моделі для заданої множини векторів стану ВБС в тому випадку, коли число зникаючих ребер не перевищує 3. Вирішення виконане шляхом декількох поетапних перетворень вихідного графу і використанні деяких положень з теорії розфарбування графів. Розглянута і вирішена задача визначення всіх попарних реберних циклів, які можуть виникнути при такому перетворенні і які можуть ускладнити саму процедуру перетворення.

ПЕРЕЛІК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Романкевич А.М., Карачун Л.Ф., Романкевич В.А. Графологические модели для анализа сложных отказоустойчивых вычислительных систем // Электронное моделирование, 2001, т.23, №1, с.102-111.
2. Романкевич О.М., Карачун Л.Ф., Романкевич В.О. До питання побудови моделі поведінки багатомодульних систем // Наукові вісті НТУУ “КПІ”, 1998, №1, с.38-40.
3. Романкевич А.М., Иванов В.В., Романкевич В.А. Анализ многоустойчивых многомодульных систем со сложным распределением отказов на основе циклических GL-моделей // Электронное моделирование, 2004, т.26, №5, с.67-81
4. Ярмолик В.Н. Контроль и диагностика цифровых узлов ЭВМ.- Минск.: Наука и техника, 1988.-240с.
5. Романкевич А.М., Гроль В.В., Карачун Л.Ф., Орлова М.Н., Романкевич В.А. Об одном подходе к расчету надежности отказоустойчивых многопроцессорных систем. // Сб. Автоматизированные системы управления и приборы автоматики. Харьков, 2002, Вып. 119, стр. 54-58.
6. Романкевич А., Ал Шбул Рабах, Романкевич В., Назаренко В. О минимизации базовых циклических GL-моделей // Вісник ТУП, частина 1, т. 2 "Технічні науки".- Хмельницький,- 2004.- С.42-45.
7. Романкевич А.М., Карачун Л.Ф., Бассем Аль Хадиди. Генераторы псевдослучайных тестовых наборов постоянного веса. // Вестник НТУУ “КПИ”. Информатика, управление и вычислительная техника. Выпуск 31, 1998, с. 34-39.

8. Романкевич О.М, Романкевич В.О., Богуславський О.В., Ал Шбул Рабах. Аналіз відмовостійких багатопроцесорних систем на основі графо-логічних моделей нециклічного типу // Вісник ТУП, т. 2 "Технічні науки".- Хмельницький, 2002.-С.30-33.
9. Романкевич О.М, Романкевич В.О., Кононова А.А., Рабах Ал Шбул О некоторых особенностях GL-моделей $K(2,n)$ // Вісник НТУУ «КПІ».- Інформатика, управління та ОТ.-2004.- №41.-С.85-92.
10. Гроль В.В. Синтез контролепригодных цифровых схем. – К.: ИСМО, 1996, 105 с.
11. Романкевич А.М., Кононова А.А. Особенности трансформации GL-моделей базовых двухустойчивых ОМС к небазовым // Радіoeлектронні і комп'ютерні системи.-№5, 2006, с.48-53.
12. А.М. Романкевич, В.А. Романкевич, Бахтари Хедаятоллах Об одном способе оптимизации моделей поведения отказоустойчивых многопроцессорных систем // Радіoeлектронні і комп'ютерні системи.- 2008.- №7.- с.49-52
13. Романкевич В.А., Морозов К.В., Фесенюк А.П. Об одном методе модификации рёберных функций GL-моделей // Радіoeлектронні і комп'ютерні системи.-№6, 2014.- С.95-99
14. Романкевич В.А., Малышева М.О., Примаков И.К. Об одном способе формирования рёберных функций GL-модели // Системный анализ и информационные технологии: материалы 18-й Международной научно-практической конференции SAIT 2016.- К.: УНК «ИПСА» НТУУ «КПИ», 2016.- С.406
15. Морозов К.В., Романкевич А.М., Романкевич В.А. О характере влияния модификации рёберных функций GL-модели на её поведение в потоке отказов // Радіoeлектронні і комп'ютерні системи.-№6, 2016.- С.108-112

16. Морозов К.В., Романкевич В.А., Сапсай Т.Г. О модификации графо-логической модели для систем и их компонентов с множественными состояниями // Системный анализ и информационные технологии: материалы 19-й Международной научно-практической конференции SAIT 2017.- К.: УНК «ИПСА» КПИ им. Игоря Сикорского, 2017.- С.94

17. Романкевич В.О. Методи і засоби оцінки технічних характеристик гарантоздатності відмовостійких багатопроцесорних систем управління складними об'єктами: Дис. на здобуття наукового ступеня доктора технічних наук: 05.13.05; - Захищена 29.01.2018; Затв. 20.03.2018.- К., 2017.- 388с.

18. Потапова К.Р., Ніколін А.Д., Ахмедова Д.Н., Алгоритм пошуку попарних реберних циклів при перетворенні GL-моделей, Прикладна математика та комп'ютинг 2018, с.123-127

19. Романкевич В.О., Ніколін А.Д., Корнейчик Б.А., Алгоритм перетворення базових GL-моделей, Прикладна математика та комп'ютинг 2018, с.141-144